



PATENT ABSTRACTS OF JAPAN

(11) Publication number: 09297684 A

(43) Date of publication of application: 18.11.97

(51) Int. Cl

G06F 9/44

(21) Application number: 09029457

(71) Applicant: FUJITSU LTD

(22) Date of filing: 13.02.97

(72) Inventor: ENOMOTO HAJIME

(30) Priority: 05.03.96 JP 08 46980

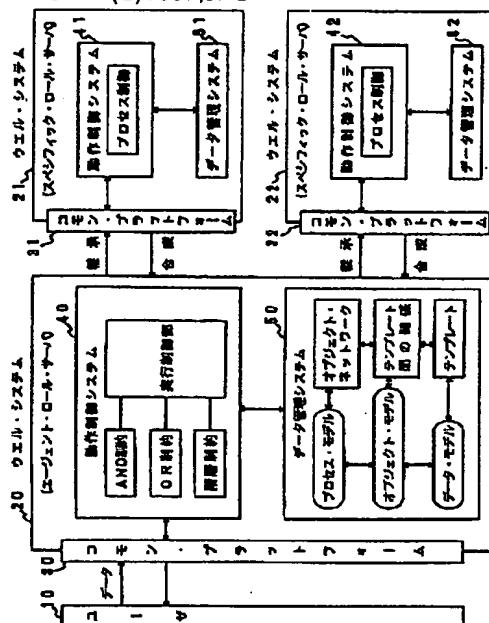
(54) INFORMATION PROCESSOR USING OBJECT NETWORK

COPYRIGHT: (C)1997,JPO

(57) Abstract:

PROBLEM TO BE SOLVED: To easily develop a visible, interactive and cooperative application by setting a data model, an object model, a role model and a process model and securing connection among these models by means of a rule that is decided on the basis of the restrictions.

SOLUTION: A data management system 50 manages a data group prescribing the hierarchical properties of a system based on the relationship among a data model which expresses an attribute value structure by means of a schema of a template form, an object model which expresses every model by hierarchically dividing the model into a form model and a feature model based on the template form of the data model, and a process model which is corresponding to an object network. An operation control system 40 checks the restrictions given to the models and secures connection among these models to advance a process based on the data group which is managed by the system 50.



(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-297684

(43)公開日 平成9年(1997)11月18日

(51)Int.Cl.⁸

G 0 6 F 9/44

識別記号

5 3 0

庁内整理番号

F I

G 0 6 F 9/44

技術表示箇所

5 3 0 A

審査請求 未請求 請求項の数12 OL (全29頁)

(21)出願番号 特願平9-29457

(22)出願日 平成9年(1997)2月13日

(31)優先権主張番号 特願平8-46980

(32)優先日 平8(1996)3月5日

(33)優先権主張国 日本 (JP)

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72)発明者 楠本 雄

千葉県船橋市上山町1丁目118番5号

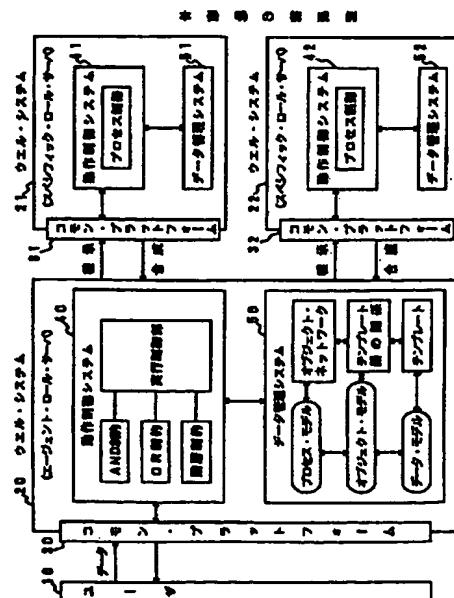
(74)代理人 弁理士 伊東 忠彦

(54)【発明の名称】 オブジェクト・ネットワークによる情報処理装置

(57)【要約】

【課題】 オブジェクト・ネットワークによる情報処理装置に関し、可視性、対話性、協調性を持つアプリケーションの開発が容易なシステムを実現することを目的とする。

【解決手段】 ウエル・システム20において、システムの階層性を、データ・モデル、オブジェクト・モデル、プロセス・モデル間の関係によって規定し、モデルの階層性をシステムの階層性と対応させる。データと制約とをシステム内で同格に扱い、データと制約の処理の流れを統一する。システム制約をオブジェクト・ネットワーク内でのオブジェクト間関数によって規定し、それによって制約処理の方法を制御する。



【特許請求の範囲】

【請求項1】 データおよびそのデータに対する処理をオブジェクトとして扱い、それらをグラフ表現したオブジェクト・ネットワークによって情報を処理する情報処理装置において、

属性値構造をテンプレート形式のスキーマを用いて表現したデータ・モデル、該データ・モデルのテンプレート形式に基づいて各オブジェクトについて形式モデルと特徴モデルとに階層的に分割し表現したオブジェクト・モデル、および名詞オブジェクトをノードとし動詞オブジェクトをプランチとしてグラフ表現され、あるノードに存在する名詞オブジェクトに動詞オブジェクトとしての関数の内容を作動させると、その動詞オブジェクト名に対応するプランチの方向にある目的対象となっている名詞オブジェクトが得られるようにしたオブジェクト・ネットワークに対応するプロセス・モデルのモデル間の関係によって、システムの階層性を規定したデータ群を管理するデータ管理システムと、

前記データ管理システムが管理するデータ群に基づいて、モデルに対して与えられた制約をチェックすることによりモデル間結合を行いプロセスを進行させる動作制御システムとを備えたことを特徴とするオブジェクト・ネットワークによる情報処理装置。

【請求項2】 請求項1記載のオブジェクト・ネットワークによる情報処理装置において、個別サービスに対応する特化されたオブジェクト・ネットワークに基づく処理を行なうスペシフィック・ロール・サーバを有し、

前記動作制御システムは、データと制約とをシステム内で同一の処理方法によって処理し、データについてはユーザーに、制約については前記スペシフィック・ロール・サーバに、サービスをリクエストし、そのリクエストに対する応答動作によりデータまたは制約を得てプロセスを進行させることを特徴とするオブジェクト・ネットワークによる情報処理装置。

【請求項3】 請求項1記載のオブジェクト・ネットワークによる情報処理装置において、前記動作制御システムは、前記制約を前記オブジェクト・ネットワーク内でのオブジェクト間関係によって規定し、それによって制約処理の方法を制御することを特徴とするオブジェクト・ネットワークによる情報処理装置。

【請求項4】 請求項3記載のオブジェクト・ネットワークによる情報処理装置において、

前記制約は、名詞オブジェクトの属性値間の制約であって必要な属性値がすべてそろわなければ次の状態へは移れないことを示すAND制約と、ある名詞オブジェクトから出発する動詞オブジェクト間の制約であって独立性を持ち並列実行が可能であることを示すOR制約と、オブジェクト・ネットワークでの処理手順上の上位オブジ

エクトと下位オブジェクトとの関係から、現時点の上位オブジェクトの属性値集合は、下位オブジェクトの新規操作によって整合化操作を受けることを示す階層制約とを含み、

前記動作制御システムは、これらの制約に基づいてプロセス実行制御を行うことを特徴とするオブジェクト・ネットワークによる情報処理装置。

【請求項5】 請求項3記載のオブジェクト・ネットワークによる情報処理装置において、

10 エキスパートがサービス実行のためのオブジェクト・ネットワークを設計するときに、システムの持つモデルに基づいて制約に関する情報を表示し、それらの制約を利用可能にする手段を持ち、その表示した制約を参照して生成された制約に関する設計に基づいて実行処理を行うプロセス・モデルが規定されることを特徴とするオブジェクト・ネットワークによる情報処理装置。

【請求項6】 請求項1記載のオブジェクト・ネットワークによる情報処理装置において、

前記データ管理システムと前記動作制御システムとによってエージェント・ロール・サーバが構成され、かつ、複数のエージェント・ロール・サーバが階層化され、

上位のエージェント・ロール・サーバが下位のエージェント・ロール・サーバをスペシフィック・ロール・サーバとみなし、

モデルの階層性によってエージェント・ロール・サーバがスペシフィック・ロール・サーバについての制御を行うことを特徴とするオブジェクト・ネットワークによる情報処理装置。

30 【請求項7】 データおよびそのデータに対する処理をオブジェクトとして扱い、それらをグラフ表現したオブジェクト・ネットワークによって情報を処理するデータ処理システムであって、

システム上で複数のプロセスが協調して動作する際に、プロセスを制御するオブジェクト・ネットワークとプロセスの状態を示す状態データとにに基づいて、イベント駆動とデータ駆動との何れかの駆動方法を該システムのカーネルが判定し、判定された駆動方法によって該複数のプロセス間の協調動作を行なながら、イベント駆動とデータ駆動との連鎖によって該複数のプロセスを進行させることを特徴とするデータ処理システム。

【請求項8】 ユーザが前記システムを操作する際に、該ユーザを前記プロセスの一つと解釈して、前記イベント駆動とデータ駆動との連鎖によって該ユーザ及び前記システム間に於ける対話を進行させることを特徴とする請求項7記載のデータ処理システム。

【請求項9】 データおよびそのデータに対する処理をオブジェクトとして扱い、それらをグラフ表現したオブジェクト・ネットワークによって情報を処理するデータ処理システムであって、

各々が所定の機能を実行処理する単位でありオブジェクト・ネットワークの構造体である複数の役割が協調動作して実行処理を行う際に、データ駆動及びイベント駆動とに基づいた通信を、該複数の役割のうちの任意の役割間で行うことを特徴とするデータ処理システム。

【請求項10】 前記複数の役割の各々は、前記通信を支援する機能を有した役割である通信支援役割によって支援されることを特徴とする請求項9記載のデータ処理システム。

【請求項11】 前記複数の役割は、主体となる役割を実行する主役割と、該主役割の動作を支援する支援役割とを含み、該主役割が動作する環境を示す環境データを、該主役割と該支援役割とが共有することを特徴とする請求項9記載のデータ処理システム。

【請求項12】 前記複数の役割の各々は、前記通信を行う機能として一対多の放送機能を有することを特徴とする請求項9記載のデータ処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、データおよびそのデータに対する処理をオブジェクトとして扱い、それらをグラフ表現したオブジェクト・ネットワークによって情報を処理する装置であって、特にシステム制約をオブジェクト・ネットワーク内でのオブジェクト間関係によって規定し、それによって制約処理の方法を制御するオブジェクト・ネットワークによる情報処理装置に関するもの。

【0002】

【従来の技術】 例えば、画像処理システムなどの特定応用分野のシステムを、オブジェクト・ネットワークとしてモデル化することによって、システム開発者にとってその特定応用分野のシステムを容易に構築できるようにし、かつ、構築された特定応用分野のシステムを用いるユーザーにとって、システムの状態や実行可能なオペレーションについて、わかりやすい形で把握できるようにした処理システムが、特開平5-233690号公報（オブジェクトネットワークによる言語処理システム）等によって知られている。

【0003】 この処理システムは、オブジェクトを、名詞オブジェクトと動詞オブジェクトとに大分類し、名詞オブジェクトをノード、動詞オブジェクトをプランチとして表現したオブジェクト・ネットワークを参照モデルとし、この参照モデルにおいて、あるノードに存在する名詞オブジェクトに動詞オブジェクトとしての関数の内容を作動させると、その動詞オブジェクト名に対応するプランチの方向にある目的対象の名詞オブジェクトが得られるような情報の処理機構を提供するものである。

【0004】 図29は、画像描画処理用のオブジェクト・ネットワークの例を示す。図29におけるNone, Point, Point seq., Line, …は名詞オブジェクトであり、

define, draw up, …は動詞オブジェクトである。例えば、ディスプレイ画面上で、画像を描画する場合を考える。最初、何もない状態であり、マウス等によって点が指定されることにより、点が描画された状態になる。さらに、いくつかの点が指定されて、点列が形成された状態に遷移する。

【0005】 このような点や点列などのデータを、名詞オブジェクトとして捉えると、ある名詞オブジェクト（例えばPoint）に、関数としての動詞オブジェクト

10 （例えばdefine）が作動することにより、目的とする名詞オブジェクト（例えばPoint seq.）が得られる。すなわち、描画対象のデータの各状態を、名詞オブジェクトとしてノードに置き、ノード間を結ぶプランチを動詞オブジェクトとして、図29に示すようにオブジェクト・ネットワークを表現すれば、そのオブジェクト・ネットワークによって、そのまま描画のための画像処理システムの構造が形成できることになる。

【0006】 また、クライアントとサーバとの対話によってオブジェクト・ネットワークに対応した処理を効果的に進めるために、オブジェクト・ネットワークに対応した可視化対象項目を表示し、かつクライアントが意図する指示とデータとを表示するとともに、サーバが実行した結果を表示する各種ウインドウを持つコモン・プラットフォームを用意し、コモン・プラットフォーム内の

20 コミュニケーション・マネージャがウインドウを介在させて、クライアントとサーバとの間におけるデータ表記形式の整合をとるなどの仲介を行なうようにしたコモン・プラットフォーム機能による対話的情報処理装置（特開平7-295929号公報）が知られている。

【0007】

【発明が解決しようとする課題】 本発明は、上記特開平5-233690号公報に開示されているオブジェクトネットワークによる言語処理システムや、特開平7-295929号公報に開示されているコモン・プラットフォーム機能による対話的情報処理装置による機能を、さらに拡張・発展させ、可視化、対話性、協調性を持つアプリケーションの開発が容易なシステムを提供することを目的とする。具体的には、制約処理をモデル概念と関連させることにより、データと制約についての規定とソ40 フトウェア構造について明らかにし、目的とする分野におけるオブジェクト間の規定を明確にし、その上でオブジェクト関係を作成すべきプロセスとの間の結合処理を行なわせる手段を提供することを目的とする。

【0008】

【課題を解決するための手段】 図1は、上記課題を解決するための本発明の構成例を示す。図1において、10は本システムに対しデータおよび指示を当たるユーザ（クライアント）、20, 21, 22はデータおよびそのデータに対する処理をオブジェクトとして扱い、それらをグラフ表現したオブジェクト・ネットワークによっ

て情報を処理するウエル (WELL : Window-based Elaboration Language) の言語系に基づくウエル・システム、30, 31, 32はウエル・システムとユーザまたは他のシステムとの対話の場としてのインターフェースを提供するコモン・プラットフォーム、40, 41, 42はオブジェクト・ネットワークに基づくプロセスを制御する動作制御システム、50, 51, 52はオブジェクト・ネットワークに関連するデータ群を管理するデータ管理システムを表す。

【0009】特に、ウエル・システム20は、ユーザ10の意図に従ってサービス計画を準備するエージェント機能を持つエージェント・ロール・サーバであり、ウエル・システム21, 22は、エージェント・ロール・サーバが準備したサービス計画を実行する役割を果たすスペシフィック・ロール・サーバである。データ管理システム50は、属性値構造をテンプレート形式のスキーマを用いて表現したデータ・モデル、該データ・モデルのテンプレート形式に基づいて各オブジェクトについて形式モデルと特徴モデルとに階層的に分割し表現したオブジェクト・モデル、および名詞オブジェクトをノードとし動詞オブジェクトをプランチとしてグラフ表現され、あるノードに存在する名詞オブジェクトに動詞オブジェクトとしての関数の内容を作動させると、その動作オブジェクト名に対応するプランチの方向にある目的対象となっている名詞オブジェクトが得られるようにしたオブジェクト・ネットワークに対応するプロセス・モデル間の関係によって、システムの階層性を規定したデータ群を管理する。

【0010】動作制御システム40は、このデータ管理システム50が管理するデータ群に基づいて、モデルに対して与えられた制約をチェックすることによりモデル間結合を行いプロセスを進行させる制御手段を持つ。ウエル・システム21, 22におけるデータ管理システム51, 52および動作制御システム41, 42も同様である。ただし、ウエル・システム21, 22は、個別サービスに対応する特化されたオブジェクト・ネットワークに基づく処理を行うスペシフィック・ロール・サーバとして動作する。

【0011】動作制御システム40は、データと制約とをシステム内で同一の処理方法によって処理し、データについてはユーザ10に、制約についてはスペシフィック・ロール・サーバ (ウエル・システム21, 22) に、サービスをリクエストし、そのリクエストに対する応答動作によりデータまたは制約を得てプロセスを進行させる。

【0012】また、動作制御システム40は、制約をオブジェクト・ネットワーク内でのオブジェクト間関係によって規定し、それによって制約処理の方法を制御する。特に、これらの制約として、名詞オブジェクトの属性値間の制約であって必要な属性値がすべてそろわなけ

れば次の状態へは移れないことを示すAND制約と、ある名詞オブジェクトから出発する動詞オブジェクト間の制約であって独立性を持ち並列実行が可能であることを示すOR制約と、オブジェクト・ネットワークでの処理手順上の上位オブジェクトと下位オブジェクトとの関係から、現時点の上位オブジェクトの属性値集合は、下位オブジェクトの新規操作によって整合化操作を受けることを示す階層制約などがあり、動作制御システム40は、これらの制約に基づいてプロセス実行制御を行う。

【0013】サービス実行のためのオブジェクト・ネットワークを設計する場合、エキスパートとしてのユーザ10に対し、システムの持つモデルに基づいて制約に関する情報を表示し、それらの制約を利用可能にする手段を持ち、その表示した制約を参照して生成された制約に関する設計に基づいて実行処理を行うプロセス・モデルを規定することができるようになっている。

【0014】ウエル・システム20のようなエージェント・ロール・サーバを、複数個階層化された形で構成することが可能であり、この場合、上位のエージェント・ロール・サーバは下位のエージェント・ロール・サーバをスペシフィック・ロール・サーバとみなし、モデルの階層性によってエージェント・ロール・サーバがスペシフィック・ロール・サーバについての制御を行う構成にすることができる。

【0015】

【発明の実施の形態】最初に本発明のシステムが扱う対象の概念について説明する。図2は、実世界と感性的対象世界との連環性を示す図である。人間が対面する実世界は、多数の物体が相互に関連し、あるいはほぼ独立に動いており、この対象世界に対し、人間は五感を働かせて、興味を懷き、想起、想像、夢として新しく発現する。この作用によって、図2に示すように実世界のごく一部が感性的対象世界に写像される。これは実世界とは異なり、仮想的なものであって、さらに対象として認知し、参照または指示可能であるにとどまる。この認知のレベルをより論理的に詳細化の階梯を上昇させることによって、人間の間でコミュニケーションがより濃密になり、興味を等しくするグループの間で協調的な意思の同化と知識の獲得が可能になる。

【0016】上記のプロセスを進行させるために、対象モデルと合目的演算モデルとが必要になる。本発明の課題は、特開平7-295929号公報 (コモン・プラットフォーム機能による対話的情報処理装置) に示されているウエル・システム (各種のオブジェクト・ネットワークに対処できる拡張性を持っているという意味で、エキステンシブル・ウエル・システムと呼ぶことがある)において、制約処理をモデル概念と関連させることにより、データと制約についての規定とソフトウェア構造について説明し、目的とする分野におけるオブジェクト間の規定を明確にし、その上でオブジェクト関係と作成す

べきプロセスとの間の結合処理を行わせる手段を提供することである。ここで、目的とする分野の実世界と感性的に認知する世界とは必ずしも一致せず、仮想表現にすぎない。したがって、ソフトウェアの作成は対話的に修正が容易である必要がある。

【0017】対象モデルは、ソフトウェアで用いられるオブジェクト概念を利用する。オブジェクトにおいて、データと制約とは、同格的取扱いが可能となる。以下、データと制約について説明する。データは関数の入力引数、出力引数として数学的に表現され、システム内では、名詞オブジェクトの属性名としてデータ名、値としては属性値が与えられる。上記ウエル・システムの体系内ではデータ定義過程に入る時、名詞オブジェクト名に対応するテンプレート(template)を用意する定義準備プロセス(definingprocess)と、値を定めテンプレート内の対応位置に値を入れる定義操作(define operation)とに段階を分割して取り扱う。動詞オブジェクトは名詞オブジェクトに対し、ある関数操作を行う。名詞格には普通名詞と有名詞の格が存在し、限定詞(a, the)によって、参照指示やデータ値の確定について一方指向性が与えられ、総称性と具体性とが決定される。

【0018】修飾については形容詞や副詞で行うが、総称性についての修飾には制約優先性によって修飾の半順序が存在する。例えば、featuredとcoloredとを比べると、この順序で制約優先性が定まる。名前 α で表現されたオブジェクトが γ で表現された性質を持つことを、内包論理学の記法に従って $\gamma\{\alpha\}$ のように結合支援記法として書く。すると、それは α で表記されたオブジェクトは γ で表記された性質を持つことを示している。従って、 $P\{j\}$ の述語の集合が $\{j\}$ というオブジェクトを規定する。

【0019】例えばLine segment(図29の[Line seg.])のリストがLineをdefineすることが、図29のオブジェクト・ネットワークに示されるような形でなされ、現在描画作業が行われているとき、painted by (the identified Colored line) = [the list of Colored line seg.]であり、対応する作業過程で現れるthe Colored line seg.が答えとして求められる。制約の性質は内包的等式で定義されるが、外延的性質を持つ要素としては、次のように考えられる。

【0020】例えばsymmetrical element(an identified element)により内包的には対称点ノ線についての相手要素の持るべき性質が与えられるのに対し、外延的には、対応する要素がデータベースより検索されなければならない。以上のように、データは具体的な属性の値として考え、制約は総称的オブジェクトの持つ性質と考える。処理システムの表現として、データはクライアントが決定し、制約はサービスとしてその内容が決定されるようにし、同格的取扱いを行う。すなわち、表記的には小括弧と中括弧の相違として処理を行う。

- 【0021】ソフトウェア・プロセスとして、ソフトウェアについての要求条件、仕様設計、実現および評価の各段階を通じて、同プロセスに関係する人々に共通に仕様可能な言語体系としては、拡張機能言語エクステンシブル・ウエル(Extensible WELL, 前述の特開平5-233690号公報、特開平7-295929号公報参照)を用いる。この言語システムの特徴は、自然言語の単語および文を用い、単語間の関係を可視化するためにオブジェクト・ネットワークをユーザとサーバとの間の共通の場としてのコモン・プラットフォームに表示し、その操作によって具体的かつ合目的な演算を行わせることである。さらに、制約について式あるいは單文形式によって表示を行い、特にプログラムを作成しなくとも、ユーザがサーバを有效地にユーザの意図通りに演算実行させることができるという特徴を持っている。
- 【0022】以上のような言語体系とインターフェクションによる協調化機能を有效地に活用するために、次のようなモデル構造を持たせる。以下、モデルの分類と構造について説明する。名詞、動詞および形容詞・副詞の修飾詞は、それぞれテンプレートを持ち、語句について修飾内容のためのテンプレートを接続して統合化が行われる。そのために、以下のようなモデルが設定される。
- 【0023】(1) データ・モデル：ウエル(WELL: Window-based Elaboration Language)の言語系では属性値構造をテンプレート形式のスキーマを用いて表現する。このデータ・モデルを通じて、オブジェクトの表記形式、データ操作などの管理を行い、データ制御を行わせ、各オブジェクトをプロセスと関連させる。
- 【0024】データの格として名詞オブジェクト、動詞オブジェクト、それらの修飾詞オブジェクト(形容詞、副詞)などを考え、既知の準自然言語形式のTELと関連させる。
- (2) オブジェクト・モデル：オブジェクト・モデルは、データ・モデルに従って形式モデルと特徴モデルに階層的に分割、表現する。
- 【0025】(a) 形式モデル：画像でいえば点、線、領域から立体構造、動き構造などをデータ・モデルとしてテンプレートにより複合形式化し、すべてのパターン項目の形式表記を行う。複数の単体間の関係、例えば前後関係による隠れや反射、屈折による新規オブジェクトもテンプレートとして表記し、形式モデル化する。すなわち形式モデルによって構文的表現が行われる。形式モデルは、画像の場合には幾何モデルに対応する。
- 【0026】動詞オブジェクトの形式モデルは、データ操作の実行開始のための preconditionとして継承すべき名詞オブジェクト名を状態制約として持つのをはじめ、制約名を指示するテンプレート部、実行中制約としての incondition、実行終了後の制約としての postconditionを持つ。これらの制約は、名詞オブジェクトに名前として付加され指示される制約と結合して、総称オブ

ジェクトから具体オブジェクトへの関係が定義される。

【0027】(b) 特徴モデル：名詞オブジェクトについての属性値をもとに特徴が表現され、それは環境についてある程度の不变性を持っている。その例は、分割線であり、顔画像の分割線で区切られた領域に目、鼻、口などの個別構成要素が配置される。ハイライト線も類似した性質を持ち、立体表現に大きな役割を果たす。

【0028】動きについての特徴モデルは、動きのための動詞オブジェクトと動き対象となる名詞オブジェクトとを中心に構成される。

(c) 感性モデル：感性モデルは人間の感性を形式モデル、特徴モデルに埋め込むためのモデルであり、感性語として名前と修飾パラメータとを随時指定し、変更しうるようとする。

【0029】(3) プロセス・モデル：プロセスとして名詞や動詞オブジェクトが表現され、オブジェクト・ネットワークとしてグラフ表現される。この設計は、クライアントのうちエキスパートによって行われ、WELLのグラフ構造エディタによってシステムが駆動可能とする。

【0030】さらにエージェント・ロール・サーバ(agent role server)とスペシフィック・ロール・サーバ(specific role server)が担当するプロセス間の協調および統合をはかるモデルが必要であり、形式モデルとしては各個別プロセス間のインタラクションの形態が定義される。なお、エージェント・ロール・サーバおよびスペシフィック・ロール・サーバの関係等については、特開平7-295929号公報に詳述されているので、ここでは簡単な説明にとどめる。

【0031】各サービスは request, respond コマンドによって継承データと制約とをもって実行依頼し、合成データと制約とが実行結果として実行報告を行うことを基本として、より効果的に協調作業を行うように役割概念を用いて形式化される。ここでも上記の形式モデルとともに、下記の特徴モデルが可能である。特徴モデルは、役割機能について環境に応じて制約機能を付加するためのモデルで、合目的演算モデルとして規定する。

【0032】前述したように、オブジェクトについての形式モデルの例として、画像の場合には幾何モデルがそれに当たる。幾何モデルを構築するためのモデルとして、スカラ・データ用とベクトル・データ用とが必要となる。その上に、対象についての構造モデルが構成されるが、ここでは、対象を構成する個別要素についての領域定義が行われる。例えば、人間の顔について言えば、目、鼻、口、耳、などの個別要素があり、それぞれが特有の支配領域を持っており、眉毛は目の支配領域の上部にあるなどの部分的制約があり、また個々の支配領域間の隣接性が制約として満足していかなければならない。

【0033】したがって、実世界から感性認知世界に写像されるのは、興味を抱き、それによって意図が駆動さ

れることによって行われ、個々の対象に始まって世界の部分領域に拡大される。この過程で、上記対象モデルが原始的データ・レベルから、図3に示すように、幾何学的に構造化される。図3は、画像についての形式モデルである幾何モデルの例を示している。

【0034】幾何モデルが設定されると、任意の静止画像を点から出発して描き、静止画像中の点が時間と共に動く様子を定義し、動画像を描画することができる。しかし、対象に関する構造を幾何モデルの上に構築しなければ、画像データベースを作成することはできない。幾何モデル中の線は、次のような構造線が不变式から導かれることが本発明者によって示されている。

【0035】

構造線 → 特徴線(尾根／谷線)

分割線(凹凸モードの分割線)

輪郭線(エッジ線)

ハイライト線

このうち、対象が変形することによる画像の変形によって生じる位相的構造が少ないので分割線であり、輪郭線も対象の外郭を表現しているならば、単一領域の特性を失わない。これらに対し、特徴線は、比較的小さい変化に対しても位相構造が変化する。これは、極大や極小点は必ず特徴線の交叉点であるが、その発生、消滅は容易に生ずることも関連している。

【0036】以上のことから、対象の外郭を切り出すエッジによって対象領域を抽出したならば、分割線によって区切られた支配領域に着目して特徴処理あるいは描画を行う。さらに動画については、多様な変形を支配領域を中心で表現する。以上のように、特徴について環境に

ある程度の不变性を持つという観点から、上述のような構造線理論によって、幾何的特徴モデルが定義される。

【0037】次に、対象の個別構成要素としての分節構造が、特徴モデルとして必要となる。すなわち、個別構成要素についての支配領域に関する不变的な隣接関係が幾何学的な動的変化をすることが、対象についての感性的特徴を表現する。感性的特徴は、個々の支配領域内および支配領域相互間の動的変化から構成されると考えると、特徴モデルは図4のように表現される。

【0038】図4は、特徴モデルの構造を説明するための図である。図4において、幾何学的表示から感性の意味表示への方向は、描画のプロセスであり、逆の感性の意味表示から幾何学的表示へのプロセスは、画像の特徴処理のプロセスであって、幾何学的表示の点や線に対して、原始的特徴を表記する特徴点やエッジ線、分割線やハイライト線の意味づけが構造線による特徴についての制約によって表記される。

【0039】幾何モデルとして、上述のことは二次元平面上の凹凸性についての特徴に関連しているが、ある属性値あるいは属性値の組についての関数の値が、空間的あるいは時間的に不变であることも、特徴値として捉え

られる。例えば、ある個体としての対象の移動についての速度がある時間領域で不变であり、その変化点が特徴を表現する。同様にある対象の輝度値の変化が、同一である領域間の境界は、エッジまたは構造拡張型エッジとしての特徴として表現される。

【0040】図2に示した対象モデルは、上記の性質、すなわち、主として制約によって駆動するプロセスを規定する。人間にとては、図4の概念構造の連鎖によって対象構造表現を理解し得るが、計算機処理を行うにあたっては、具体的実行を計算機に合目的となるように合わせるために必要な合目的演算モデルを必要とする。以上のようなオブジェクト・モデルとプロセス・モデルとの結合は、対象構造と実行処理の両表現間の簡単な変換プロセスを必要とする。このようなモデル間結合は、次のように制約についての規則について規定し、規則集合に対する対話的制御システムによって全システムを効果的に動作させることができる。

【0041】次に、モデル間結合について説明する。プロセス実行処理は、データ・モデルをもとにして行う。データは原則としてクライアントから与えられ、制約の内容はモデル内で定義されているものと、クライアントが環境に応じて規定するものとがあるが、その種類としては次のものがある。

【0042】(a) AND制約：名詞オブジェクトの属性値間。ある名詞オブジェクト名への複数個の動詞オブジェクト間には次の制約規則がある。

(b) OR制約（独立性）：ある名詞オブジェクト名に属する複数項目への指示間、ある名詞オブジェクト名から出発する動詞オブジェクト間。

【0043】(c) 階層制約：オブジェクト・ネットワークでの処理手順上の上位オブジェクトと下位オブジェクトの関係から、現時点の上位オブジェクトの属性値集合は、下位オブジェクトの新規操作によって、整合化操作を受ける。ある動詞オブジェクト操作によって名詞オブジェクトの属性値を変化させたとき、それより上位の名詞オブジェクト群について整合化修正を行う。

【0044】(d) 優先制約：修飾詞間の優先規定、プロセスなどにおける役割機能の優先性を規定する。プロセスの進行過程は、インタラクションとしての対話過程を考えると、プロセス動作の可視化→オブジェクト・ネットワーク、クライアントとの対話性→コモン・プラットフォームに情報集約化、クライアントのユーザとエキスパートの役割分担、協調過程の効率的表記→役割機能の制御、を満足させる形でモデル間結合を行わせることになる。そのために、コモン・プラットフォームに対してユーザと関係するデータと制約とが与えられ、表示されているオブジェクト・ネットワークで進行が管理される。

【0045】図5は、本システムにおける実行処理の進行の例を示している。オブジェクト・ネットワーク内の

実行処理は request-respondベースで行うが、その相手先でプロセスからデータまでの継承および合成についての制御規則が、プロセスに関する優先性を含めた形で、図5のように管理される。以上のようにモデル間結合は、データ、オブジェクトおよびプロセスの各種モデル間の制約規則をもとに行われ、それが具体的に実行管理を進める。

【0046】オブジェクト・ネットワークの例として、図29に示す画像描画処理用のオブジェクト・ネットワークの場合をより詳しく説明すると、Colored pointをマウスなどでクリックすると、同ネットワークのテンプレートの規定によって、図6に示すように、データ・ウインドウ上に輝度値用バー、輝度勾配ベクトル、色度ベクトルを指定するためのデータ・ウインドウが現れる。

【0047】この事実は、名詞オブジェクトの属性値間でのAND制約は、ある値としてidentifyされた Colored pointの名前のオブジェクトについては、輝度値、輝度勾配ベクトル、色度ベクトルがすべて完備して初めてこの名詞オブジェクトのデータが完備すべきことを規定している。このことをある Colored pointについて考えると、まず Point項目に属する点について、identifyに相当する点指示操作が行われることを前提として Colored point処理に移る。

【0048】また、OR制約（独立性）は、Pointの指示を何回か独立に行ってもよいこと、および Pointから Colored pointへのidentifyとAND制約下に属性値表示を複数回独立に行ってもよいことを規定するもので、これらの制約についての操作上の規定は、ユーザの持つ一般常識と一致し、何らのユーザ操作を必要とせず、オブジェクト・ネットワークのユーザフレンドリ性をシステムとして規定するものである。

【0049】階層制約は、オブジェクト・ネットワークで Colored regionまでユーザ操作が進んだとき、Colored regionについてのテンプレートは、各水平走査線、すなわち Colored region segmentのテンプレートにおいて、規定された Colored lineのテンプレートから、図7に示すように Lineが着目している水平操作線上の座標値 (X, Y)、輝度値 L、輝度勾配ベクトル (Lx, Ly)、色度ベクトル (Cx, Cy) を横座標 Xについてリスト構造化する。

【0050】この段階で、下位オブジェクトが新規操作によって追加もしくは削除されたとき、図7のテンプレートの内容が変化し、新規あるいは削除された X の値によって、上位オブジェクトの構造が変化し、新しく名詞オブジェクトの階層間に整合化修正を行う。このためにシステム内で階層化制約規則が駆動され、上記の整合化処理を行う。

【0051】以上の制約規則は、本発明において、拡張機能言語 Extensible WELL の体系自体に組み込まれたシステム制約であり、与えられたオブジェクト・ネットワ

ークに対して、制約に基づいたユーザフレンドリな操作環境を与える。すなわち、オブジェクト・ネットワーク上で現在処理している名詞オブジェクト名が状態としてシステムを規制する。この状態の変化が制約規則との関係を指定する。この状況を図8に示す。

【0052】図8は、システム制約とWELLとの関係を示しており、図8(A)は実行時システム制約における実行制御、図8(B)はオブジェクト・ネットワーク設計時の制約規則による対話制御の例を示している。ユーザは、図8(A)に示すように、コモン・プラットフォーム30を介してウエル・システム20に対する指示やデータの入力を行う。コモン・プラットフォーム30として、オブジェクト・ネットワークを表示するオペレーション・ウインドウ61、データの表示を行うデータ・ウインドウ62、システムのメッセージを表示するメッセージ・ウインドウ63があり、これらによって、ユーザが意図する指示とデータ、およびシステムが実行した結果が表示される。

【0053】ウエル・システム20は、オブジェクトの状態の履歴を記憶するオブジェクト状態ヒストリ71および現在操作対象となっている現・操作オブジェクト72について、実行制御部76によって、AND制約73、OR制約74、階層制約75のチェックを行い、これらの制約チェックに基づいてプロセスの実行を制御し、データ記録部77によって実行結果の記録を行う。

【0054】オブジェクト・ネットワーク設計時には、図8(B)に示すように、コモン・プラットフォーム30によって、オブジェクト・ネットワークの表示と、参照している修飾詞オブジェクト(形容詞や副詞)間の接続構造の表示を行い、エキスパートがこれらのシステムの持つモデルに基づいて制約を利用できるようにし、ウエル・システム20によって、その制約に関する設計に基づいて、修飾詞の優先順位のデータベースを作成したり、オブジェクトのテンプレート表を作成したりすることによって、実行処理を行うプロセス・モデルを規定することができるようになっている。

【0055】図9は、実行時システム制約(AND制約)による動作制御システムに関連する処理の流れを示す。AND制約では、名詞オブジェクトの複数の属性値を次の処理によってユーザに指定させる。

(1) ユーザは、コモン・プラットフォーム上での操作により、名詞オブジェクト名を指示する。

【0056】(2) ウエル・システムは、定義準備プロセスとしてAND用マークをof状態にしたテンプレートを設定する。そして、AND条件を設定し、データウインドウの開設表示を行う。

(3) ユーザは、コモン・プラットフォーム上でのデータ・ウインドウにおいて、属性値の定義操作を行う。

【0057】(4) ウエル・システムは、定義データをテンプレート上の対応位置に格納し、内容を表示する。ま

た、対応テンプレート部のADN用マークをon状態に設定する。

(5) AND制約をチェックし、すべての属性値がそろったならば次のプロセスに進む。まだそろっていないければ、データ定義待ちとする。

【0058】図10は、実行時システム制約(OR制約)による動作制御システムに関連する処理の流れを示す。OR制約では、次に選ぶ動詞オブジェクトをユーザが自由に選ぶことができる。

10 (1) プロセスの進行に伴い、ウエル・システムは、実行済みオブジェクトに複数の動詞オブジェクトが存在するかを調べ、複数の動詞オブジェクトについて任意の動詞オブジェクトを選択可の状態にする。

【0059】(2) ユーザは、次の動詞オブジェクトを選択する。図11は、階層制約を説明するための階層の例を示す。オブジェクト・ネットワークの進行過程において、上位オブジェクトについて処理がすでに行われている状態で、下位オブジェクトの選択あるいは消去が、その下位オブジェクトの指示によって行われるとき、そのより上位で定義済みの上位オブジェクトの属性値データを設計し、整合化する必要がある。

【0060】このような階層制約を実現するために、オブジェクト・ネットワークが図11に示すような階層構造であって、四角印で示すように、局部的最上位オブジェクトとして定義済みのオブジェクトA, Bの状態表示が行われていたとする。このような場合、三角印を付加したオブジェクトFの属性値を設計するときには、オブジェクトFの属性値の修正は、既に設定されているテンプレートの構造はそのままで、内容を四角印のオブジェクトA, Bまでの、オブジェクトC, D, Eの属性値を設定しつつ、オブジェクトA, Bの属性値を設定するような設計に進む必要がある。

30 【0061】また、二重三角印のオブジェクトHの新設もしくは消去が行われるときには、そのオブジェクト名に属するオブジェクト名を付加するか、または指示された特定のオブジェクトを消去する。これらのとき、オブジェクトが上位のオブジェクトA, Bに至る経路にあるテンプレートの内容構造を変更する準備を整えることが、整合化操作として必要になる。

40 【0062】以上のために、ウエル・システムの実行制御部に、現在処理中のオブジェクト・ネットワークにおいて、実行済み局部的最上位オブジェクト名を状態として記憶する局部的最上位状態記憶部と現実行オブジェクト名を記憶する現・状態記憶部が存在する。図12にその階層制約処理の流れを示す。図12中の上位でのトレース操作は、整合化のための各オブジェクトに対するテンプレート内容を設定するため、定義準備と定義操作を行うべきことを意味する。このため、オブジェクト・ネットワークで設定が未完了であるときには、メッセージ表示をコモン・プラットフォーム上に行う。

【0063】次に、本発明に関するオブジェクトの品詞について説明する。拡張機能言語Extensible WELLにおいては、すべてのサービスが“User puts a point.”などのように述語形式により表現され、それが実行されており、扱われるオブジェクトの品詞によりその用途が異なる。このことから、各オブジェクトに対して品詞を区別するためのパラメータをテンプレートに付加する。これにより、そのオブジェクトがサービスの状態を指すのか、動作を指すのか、それらの制約を指すのかを区別することが可能となる。

【0064】サービスの形態が述語形式で表現されるため、各オブジェクト間の接続形状によってサービスの内容が異なる。例えば、動詞に対する目的語（動詞+名詞句）、動詞に対する副詞（動詞+副詞）、形容詞に対する名詞句（形容詞+名詞句）等がある。これらに対応するオブジェクトに対しての接続可能なオブジェクトには、優先性が存在する。これらの優先性を確定させるために優先順序を必要とするオブジェクトに対しては、優先順位表を参照し、接続順序を決定する。

【0065】図13は、オブジェクトの接続順序の例を示している。例えば、図13に示すように、coloredとfeaturedという形容詞オブジェクトがあり、これらがpointという名詞オブジェクトを修飾するとき、優先順位表を参照して、まずcoloredをpointに接続し、featuredをcolored pointに接続する。

【0066】また、階層制約により上位の階層の内容よりも下位の階層の内容が優先される。これは、オブジェクト・ネットワーク上で何らかの修正等が発生した場合において、それより上位の階層の内容に影響するということである。これにより、オブジェクト・ネットワーク上で、上位階層に存在するオブジェクトよりも、下位階層に存在するオブジェクトによって整合化調整される。

【0067】各オブジェクトのテンプレートおよびデータスキーマについて、以下に説明する。現在、拡張機能言語Extensible WELLにおいては、名詞、動詞、形容詞、副詞の4つの品詞が存在する。それぞれのテンプレート形状をここで示す。これらの内容は、そのオブジェクトを設定するエキスパートによって決められるものである。

【0068】【名詞オブジェクト】名詞オブジェクトにおけるテンプレートにおいて必要な要素としては、名詞オブジェクト名、品詞を示すパラメータ、時相制約(temporal constraint)を示すconstraint用pointerと、そのオブジェクトの任意のテンプレートである。また、その他にこの名詞オブジェクトに対して接続可能な形容詞オブジェクトの優先順序を示した優先順位表が存在し、それを参照して接続形状を判別する。

【0069】【動詞オブジェクト】動詞オブジェクトは、動詞オブジェクト名、品詞を示すパラメータ、そのオブジェクトのインプットデータ（実行前の状態の名詞

句）、アウトプットデータ（実行後の状態の名詞句）、この動詞オブジェクトがどのような用途で用いられるかを示したテキスト文（これは、メッセージ・ウインドウを介してユーザに対してその用途を表示するためのものである）、時相制約(temporal constraint)および形態制約(model constraint)を示すconstraint用pointerによって構成される。

【0070】図14に、動詞オブジェクトのテンプレートとデータスキーマの例を示す。図14のテンプレートにおいて、動詞オブジェクト名は、このオブジェクトの名前である。品詞IDは、このオブジェクトの品詞を表すパラメータであり、例えば名詞は“0”、動詞は“1”、形容詞は“2”、副詞は“3”となっている。input dataは、インプットデータのオブジェクト名を指定し、output dataは、アウトプットデータのオブジェクト名を指定する。textは、このオブジェクトの使用方法の説明文、constraintは、constraint用pointerを表す。

【0071】総称的動詞オブジェクトにおいては、同一名称でありながらその用途が異なることがある。この場合、システムの状態（インプットデータとアウトプットデータの積表現、もしくは副詞等の制約）によって、総称的オブジェクトから状態オブジェクト、さらに具体的オブジェクトへと変化し、サービスの内容を決定する。これは、プロセス構築層によって決定され、動作制御システムによって管理される。

【0072】【形容詞オブジェクト】形容詞オブジェクトは、形容詞オブジェクト名、品詞を示すパラメータ、constraint用pointerとその内容を示す任意のテンプレートによって構成される。また、これに対応する優先順位表を参照し、名詞オブジェクトとの接続順序を決定する。

【0073】【副詞オブジェクト】副詞オブジェクトは、副詞オブジェクト名、品詞を示すパラメータ、constraint用pointerによって構成される。また、これに対応する優先順位表を参照し、動詞オブジェクトとの接続形状を確認する。以上述べたのは、WELL中に埋め込まれたシステム制約であるが、このほかに、ユーザが指定するサービスに関する制約がある。

【0074】次に、サービス分割について説明する。有用なエージェント機能を実現するためには、複雑なサービスを細かく分割していくことが必要になる。その分割は大きく分けて2種類に分類される。層分割と並列分割である。層分割はネスト形式で表現され、3つの状態、プレ、イン、ポストを持つ。プレ状態においては必要なプリプロセスが実行され、イン状態では次の層に対する実際のサービスが行われ、ポスト状態では行われたポスト結果の妥当性検査が行われる。

【0075】並列分割は、サービスの相互排他性のあるものに対して行われる。分割されたサービスは並列実行

が可能である。制約分類は、以下のとおりである。分割されたサービスは、それぞれ制約を用いて管理される。層分割に対しては時相制約、並列分割に対して形態制約が用いられる。

【0076】時相制約では、必要な条件がAND制約として扱われる。すなわち必要な条件の全てがそろわなければ、次の状態へは移れない。図29に示すオブジェクト・ネットワークにおいて、ある点を画面上に書くという単純なサービスを考えてみる。このサービスは、拡張機能言語Extensible WELL 上では、「draw up Point」として表現される。「draw up」という動詞オブジェクトに対して、「Point」という名詞オブジェクトの目的語が付随する。この「draw up Point」の例における点の座標データは、この操作における時相制約といえる。独立性を持つプロセスには、OR制約（独立性）が当てはまり、並列実行が可能となる。

【0077】形態制約については、上記並列分割された際に、独立に出されたサービス結果を統合する場合に用いられる。例えば、相互独立に書かれた画像を合成する場合、それぞれの相対的な大きさや向き、角度などがそれにあたる。また、プロセスに関する制約もこれに含まれ、階層制約という。ある下位オブジェクトの新規操作をするとそれに伴って上位オブジェクトにおいても整合比調整をする必要がある。さらに前述の役割機能の優先性についての制約として優先制約がある。これはプロセスにおける最優先当事者を決定するのに用いられる。

【0078】制約処理の実現は、以下のように行う。上記のような制約条件を用いて制約処理を行う場合、本言語でこれまで用いられてきた定義準備(Defining process)と定義操作(Define operation)を拡張して管理を行う。定義準備において対応するテンプレートを準備し、定義操作においてデータをテンプレートに格納するという基本概念は継承する。ただ、制約処理を実現するためには、テンプレート自身が動的に変更可能である必要がある。つまり、テンプレート群、テンプレートリストを管理する必要がある。動的に変更されるテンプレートを決定するのが制約条件である。

【0079】図15は、制約処理と内包論理を説明する図である。拡張機能言語Extensible WELLにおいて、データは値指向で定義される必要があり、その値は既に定義されていなければならない。一方、制約については名称指向であり、その値自身は定義されていない。名称参照のみが可能で、図15に示すように、内包関数に可能世界を与えることによって外延となり、値が定義され、データと等価になる。上記の動的なテンプレートの変更もこの内包関数のために必要になる。

【0080】先に述べたデータと制約を統一的に管理するという方法論は内包論理式によって実現される。データは $\alpha(\beta)$ で表現され、制約は $\alpha(\beta)$ で表される。この $\alpha(\beta)$ は、 α についての性質を述語表現したP

(j) の集合として記述される。オブジェクト β は属性 α を持っている。() は可能世界を与えることによって属性 α を満たすようなオブジェクト β を生成することを表す。この生成が制約処理であり、2種類の生成方法がある。データベース検索とプロシージャである。

【0081】データベース検索の例としては、Highlight Point の抽出がある。指定された水平走査線上の点の中で、エキスパートによって指定された、ある一定以上もしくは最大のハイライト値を持つものを抽出する。この場合のHighlight Point は名詞であるPoint を修飾しているので形容詞と捉えることができ、ある特定のPoint に対する限定詞として使われている。

【0082】プロシージャの例としては、先のあるオブジェクトに対して対称に点を打つ例が当てはまる。この場合、オブジェクトPoint は、あるオブジェクトに対して対称であるという属性を持ち、エージェント・ロール・サーバはその条件を満たすPoint を生成するために必要なプロシージャを呼ぶ。エージェント・ロール・サーバは、この作業に対する最優先当事者を知っているので、そちらへ依頼することになる。この場合の最優先当事者が人間であるユーザーの場合と、機械であるサーバの場合とでは、インターフェースが異なる他は等価とみなされる。つまり、ユーザーもサーバの一つであるということである。

【0083】こうして求められた値は、定義操作として決定されたテンプレートに格納され処理が進んでいく。サービス制約は、あるオブジェクトの属性値間あるいは複数のオブジェクトに関して属性値間の関係として表現される。例えば、2本のLineの対称性は、対称線に対して2本のLineが鏡像となっていることを要求するものである。この場合には、対称性に関する制約をシステム内の制約データベースから抽出するか、あるいはユーザーが制約関係式としてオブジェクトの属性値を用いて定義する。この場合、この関係を満足させるサーバに要求する。データの場合、前述のようにユーザーがサーバとなっている。したがって、制約の場合もデータと同じく、定義準備と定義操作がサーバに対して設定される。

【0084】サービス制約についてのこの状況は、特開平7-295929号公報の図14、図15、図16に説明しているものと同様である。また、同公報にエージェント機能によるコモン・プラットフォーム機能の合目的的平行実行機構として示されているウエル・システムの支援と制約処理にも関連している。以上のように、データ・モデル、オブジェクト・モデル、プロセス・モデルが階層性を持ち、これらのモデル間結合の存在によって、ユーザーの意図が計算機による具体的実行に変換される。この変換過程が合目的演算モデルであり、具体的には、プロセス・モデルとして表現される。このプロセスについてのユーザーの意図は、コモン・プラットフォーム上に表現されたオブジェクト・ネットワークについての

操作によって、ウエル・システムによって実現される。

【0085】したがって、プロセス・モデルは、エージェント・ロール・サーバ(agent role server)とユーザとの間でコモン・プラットフォームによって演算プロセスが制御され、エージェント・ロール・サーバ(agent role server)は、スペシフィック・ロール・サーバ(specific role server)との間でサービスのrequestを行い、実行結果が respondとしてエージェント・ロール・サーバに返される構造になっている。

【0086】この構造は、例えば図16に示す階層化されたエージェント・ロール・サーバのように、エージェント・ロール・サーバを階層化して構成し、上位のエージェント・ロール・サーバが下位のエージェント・ロール・サーバをスペシフィック・ロール・サーバとみなすような構成にすることも可能である。以上のサーバ間に存在する対話過程は、(a) 可視化、(b) 対話性、(c) 協調性を保つように特開平7-295929号公報に示されている考え方によって実現される。

【0087】モデル間結合方式の実現では、ユーザのコモン・プラットフォーム上のオブジェクト・ネットワーク上に表示されたオブジェクトの操作において、名詞オブジェクトが指示されると、そのデータについてのモデルとして、まず属性値構造に基づくテンプレートが必要で、そのために、次の処理を行う。

(a) 定義準備プロセス(defining process)によるテンプレートをウエル・システムの作業領域に呼び出し、データ・ウインドウを設定する。

【0088】(b) コモン・プラットフォーム上に設定されたデータ・ウインドウをユーザが指定し、データ値を定義する。この段階が定義操作(define operation)である。ここで、名詞オブジェクトの属性値間に性質としての制約が存在し、その制約を満足するようにするために、既定義済みの属性値と未定義の属性値を求めるプロセスを起動する必要がある。ここにおいて実行しているエージェント・ロール・サーバが、制約を満足する解を求める役割を担当しているスペシフィック・ロール・サーバにパラメータとしての属性値集合と制約を規定する性質を与える、解を求めるようにrequestする。スペシフィック・ロール・サーバは、前述のエージェント・ロール・サーバから与えられたパラメータをもとに、制約についての解を respond結果として返す。

【0089】以上のこととは、(a) データが、ユーザとエージェント・ロール・サーバ間の通信に相当し、(b) 制約が、エージェント・ロール・サーバとスペシフィック・ロール・サーバ間の通信に相当する、とみなすことが可能で、ウエル・システムとしては、ユーザもスペシフィック・ロール・サーバの一種で、ユーザの意図を満足するようユーザ自身が解を与えると考えられる。したがって、ウエル・システムとしては、データ/制約について同じプロセス・モデルに基づいて実行を進めることができる。

可能である。

【0090】例として、連續性制約によって画像修正を行う例を述べる。画像の連續性保障と制約駆動は、次のように行われる。

【画像の連續性について】任意画像上に存在する物体の形状を変化させる場合、その部分を領域という形で指定して、領域を形成する各主要点に移動ベクトルを与える。これによって領域内部は、連續的な変形を行うことができる。しかしながら、そのまわりでは変形によって

10 その特徴が大きく変化する。具体的には、領域が移動してしまったり、領域が移動してきたことにより、移動が起こらなかった場合との連續性が失われ、明らかに領域が移動してきたと分かる不連続な部分が現れる。これは、領域の変形という条件は満たされていても画像の連續性という点では満たされておらず、視覚的にも不自然な画像となってしまう。

【0091】図17は、領域の移動に伴って画像の不連続な部分が生じる例を示している。実線が移動前の領域であり、この領域が破線に示すように移動することによって不連続な部分が現れる。このような領域を持つ画像の連續性を保障するためには、システム側がこれらの現象を防ぐ処理を行う必要がある。そのため制約駆動を用いる。

【0092】【制約駆動について】制約駆動とは、データの不備に対して制約による代用データを用いた駆動形式のことをいう。例えば、ある領域を移動させると、その移動量が大きい場合には周辺部分との連續性が失われる。連續性が失われた時点でユーザに修正領域を指定させるのではなく、システム側があらかじめ修正領域を確保しておき、その領域に基づいた処理を実行する。この時点で処理結果がユーザの満足するものであれば、次の処理へ移行する。しかしながら、その処理結果が必ずしもユーザの満足するものであるとは限らない。そこで再処理を行ったり、ユーザに修正領域を指定してもらうといった処理の変更を行う。これは、複数の処理方法があるため、ある一つの処理方法では継続が不可能でも、別の処理方法に切替えることで継続が可能になることを示している。

【0093】オブジェクト・モデルの形式モデルは、前記のデータ・モデル中のテンプレート構造を属性値構造として表記する。これは、ある場合にはデータとして、ある場合に制約として表現される。オブジェクトの特徴モデルとしては、一般的にある制約として構造線のような幾何学的特徴があり、それらの分節構造が、画像中の領域間関係で表現するときは感性モデルとも関係し、ユーザの指示操作がデータ・モデルのように基本的操作を通じてシステムを制御するのが一般的である。

【0094】以上のこととは、制約については階層性が存在することを意味する。例えば、前述の連續性制約が支配領域間関係に対し与えられているとき、支配領域の変

形がまず必要で、そのためには支配領域を定義する特徴点が与えられ、それから領域定義を行う階層構造が必要である。図18は、画像の階層制約処理の例を示している。

【0095】図18の処理によって、図29に示すオブジェクト・ネットワークの要素ネットワーク上で領域(Region)からその領域上の点Pointについて特徴点を変更し、新領域を得るが、属性値を元の値から付与すると、連續性に不都合を生じるので、この不連續性を解消するための修正を行う。このような階層にまたがって実行処理を行うとき、制約処理がウエル・システムによってシステム制約として行われ、画像修正を新しく行うプロセスが起動される。

【0096】上記の内容は、システムの階層性がモデルの階層性と一致し、モデルによって抽象的オブジェクトから具体的オブジェクトに変換する方法が規定されることを意味し、その逆変換も同様である。また、抽象性を持ったオブジェクトによってユーザの意図についてエージェント・ロール・サーバが対話をを行い、それをパラメータによって具体化がエージェント・ロール・サーバの仲介を行って、スペシフィック・ロール・サーバがユーザと全システムとしては間接的に通信を行う。しかし、処理の中身については、ユーザには具体的要求となるよう実現される。

【0097】図19は、動画像描画システムにおけるサービス制約の一環である連続性制約に関する対話的処理の流れを示す。

(1) ユーザは動画像の描画を意図した場合、コモン・プラットフォームを介して、動画像の1単位となる静止画の要求操作を行う。

(2) エージェント・ロール・サーバは、要求画像名によるデータベースの検索操作のためにスペシフィック・ロール・サーバを呼び出し、スペシフィック・ロール・サーバは、対応画像名データを抽出して検索処理を行う。

【0098】(3) スペシフィック・ロール・サーバは、要求画像名に対応する画像データを集積し、エージェント・ロール・サーバに返す。

(4) エージェント・ロール・サーバは、中継操作を行い、コモン・プラットフォームを介して要求された単位画像を表示する。

(5) ユーザは、表示された画像について凹凸分割線などにより支配領域へ分割することを要求する操作を行う。

【0099】(6) エージェント・ロール・サーバは、コモン・プラットフォームを介して支配領域抽出の要求を受け、スペシフィック・ロール・サーバを呼び出す。

(7) スペシフィック・ロール・サーバは、対応画像について特徴処理のための構造線、制約式をデータベースから抽出し、制約から分割線処理の実行を要求する。

【0100】(8) エージェント・ロール・サーバは、中継操作を行い、コモン・プラットフォームを介して、対

応画像に特徴点表示を含む支配領域データの表示を行う。

(9) ユーザは、特徴点の組の指示による部分支配領域の指示と特徴点の移動量の指示による部分支配領域の変形指示を行う。このとき、コモン・プラットフォームによる部分領域の変形方法を指定するためのベクトル表示等を介した指示がエージェント・ロール・サーバに伝達される。

【0101】(10) エージェント・ロール・サーバは、担当部分領域の指定と変形をスペシフィック・ロール・サーバに要求する。

(11) スペシフィック・ロール・サーバは、個々の部分支配領域の変形処理の依頼を受けて、変形処理を実行する。具体的には、個々の部分支配領域について変形方法として指定された動きベクトル等に基づき、部分支配領域における各画素についての移動等を行う。

【0102】(12) エージェント・ロール・サーバは、部分支配領域の統合のため、コモン・プラットフォームを介して変形前後の支配領域を表示する。

【0103】(13) ユーザは、隣接支配領域の連続性制約条件の評価とその適用を要求し、適用に必要なパラメータを入力する。

(14) コモン・プラットフォームは、連続性制約についてのパラメータをエージェント・ロール・サーバに伝え、エージェント・ロール・サーバは、隣接領域の連続性制約処理を駆動するためのスペシフィック・ロール・サーバの呼出しを行う。

【0104】(15) スペシフィック・ロール・サーバは、隣接領域における連続性制約による修正を行う。個々の部分支配領域についての修正量を算出し、エージェント・ロール・サーバに伝える。

(16) エージェント・ロール・サーバは、修正量の統合を行い、コモン・プラットフォームを介して修正画像を表示する。

【0105】(17) ユーザは、変形画像が受け入れ可能かどうかをチェックし、可能であれば承認する。受け入れ不可能であれば、再処理を指示する。

(18) 以上の処理を繰り返し、逐次、動画像を描画していく。

【0106】(19) すべての描画動画像が認証されたならば処理を終了する。前述した説明と一部重複する部分があるが、次に、拡張機能言語Extensible WELL を用いたシステムにおける対話的動作制御の方式について説明する。

【0107】本システムの実現にあたっては、多種類のサービスに対応し、ユーザとシステム間およびサービス・プロセス間の協調的作業を円滑かつ効果的に進めるために、インターフェースの役割をするシステムが必要である。すなわち、処理システムの動作状態を的確に監視し、表示し、制御するシステムが必要である。以下、このシステムを対話的動作制御システムと呼び、その構成

を明確にする。

【0106】この対話的動作制御システムのサービスの対象は、以下のとおりである。本ウエル・システムでは、協調処理を対話的に行うことによって効果を上げよう意図しているが、そのため必要な動作制御システムのサービスの対象は、大きく分けて二つ考えられる。ユーザがサービスを要求する際と、複数のスペシフィック・ロール・サーバで協調処理を行わせる際である。前者の動作制御システムの制御の流れを図20に示し、後者の動作制御システムの制御の流れを図21に示す。

【0107】それぞれの動作制御は、二つの制約処理、すなわちユーザとの対話によって作業内容を制約する形態制約(modal constraint)と、作業状態の完了のチェックなどを行う時相制約(temporal constraint)を対話的に行うことによって実現できる。まず、ユーザによるサービス要求時の動作制御を説明する。

【0108】ユーザがサービスを要求する際に、的確に作業を進めるために、操作を正しい方向に導くことが必要となる。例えば、動画像生成過程で、未だ静止画像を描いていない状態で絵を動かす要求を誤って出しても、動かす絵がないので、その要求をどこに出せばいいかわからないことになりかねない。そこで、ユーザの要求が確実に実行できるサービスに要求意図を限定する必要がある。

【0109】この動作制御システムでは、ユーザが依頼したサービスをもとに、総称的データ管理システムやプロセス構築層によって、作業を行うプロセスの準備を行う。動作制御システムは、現在のサービスの作業進行状態を把握するように構成できるため、可能なプロセスだけに実行許可を出す。これは、動作制御システムの形態的制約処理を行う部分である。なお、サービスの作業状態はフラグを立てることにより、把握することができる。

【0110】サービス準備のチェックを行う際の動作制御は、以下のとおりである。ユーザがサービスを要求する際に必要な制御として、ユーザが指定したサービスを実行する準備が本システムにできているかのチェックを行うことも必要である。本ウエル・システムでは、前述のように、各分野に対応する処理プロセスにおいて、状態を名詞オブジェクト(ノード)、状態遷移の関数を動詞オブジェクト(プランチ)とするグラフ構造で表現した処理形態を、オブジェクト・ネットワークとして定義し、オブジェクト・ネットワークをもとにサービスを実現する。

【0111】各サービスを実行しようとする際に、このオブジェクト・ネットワークの起動が完了しておらず、作業ができない場合、プランチの処理が終了しておらず、次のノードへの作業ができない場合など、サービス実行が不可能な場合を考えられるので、動作制御システムがそれらをチェックし制御することになる。これらの

制御について、準備完了ならばフラグを立てることにより、チェックを実現できる。これは動作制御システムの制約処理の一つで時相的な制約処理を行う部分である。

【0112】サービス実行時のデータが不十分であるときの動作制御は、以下のとおりである。ユーザが要求したサービスを実行させるための制約が十分であるかをチェックする必要がある。例えば、動画像の例で二つの物体を衝突させる作業を行うサービスを考えると、物体をそれぞれ抽出させた後、それぞれの大きさ、動きデータ等の制約条件が必要となり、それらの制約条件がないと衝突させる処理を行うことができない。衝突させようとする際のイベント駆動で制約が不足していることがわかると、ユーザもしくはサービス・モジュール(それぞれの分野に対応する処理を行う関数の集合)に要求することになる。もちろん実行するのに必要なデータがすべてそろっていると、動作制御システムは呼び出されることがない。

【0113】一般に制約条件のチェック方法は、動詞オブジェクトからデータ管理システムにそのサービスを実行するのに必要な制約条件を問い合わせ、それをもとにチェックを行う方法をとる。階層制約における動作制御については、以下のように行う。拡張機能言語Extensible WELL による画像描画システムのオブジェクト・ネットワークに表現されているサービス・プロセスに基づき、上位にあるオブジェクトと下位にあるオブジェクトの関係から、下位のオブジェクトの新規操作によって上位のオブジェクトが修正される階層制約処理を行う必要がある。上位オブジェクト修正の際にデータ(制約)不足になる場合が生じるが、動作制約システムはデータ不足と認識し、ユーザもしくはサービス・モジュールと対話することによって必要なデータを要求することになる。

【0114】図22は、動作制御システムによる制御の例を示す。以上説明した動作制御システムを用いて実際に制御する例を、図22に従って説明する。この例は、顔画像など画像の変形に伴い生じる画像の不連続性を修正するオブジェクト・ネットワークで作業を行う場合の例である。作業に必要な制約を要求する処理において、制約作成の当事者はユーザの例である。動作制御システムは、図20で説明したようにユーザが実行できるサービスを現在の状態から限定する動作制御を行った後、顔画像の表情変化をさせるサービスを選択した場合で考える。

【0115】このオブジェクト・ネットワークは、顔画像中のいくつかの領域を変化させ、隣接領域との関係を見て不連続とならないように設定した制約の範囲内であるかどうかを判断することになる。ここで、移動量が少ないなど、変更が制約の範囲内であると、次の作業を行うことになる。しかし、制約の範囲外であることがわかると、画像の不連続などの問題が生じ、画像の凹凸によ

る濃淡を分離している分割線によって生成される領域である支配領域の修正が必要になる。

【0116】制約の範囲外であると、オブジェクト・ネットワークは、動作制御システムに制約処理、つまり修正が必要であることを伝える。動作制御システムは、領域変更を行うオブジェクト・ネットワーク中の動詞オブジェクト（プランチ）から判断して、支配領域を修正するオブジェクト・ネットワークへ作業要求を出すことになる。この際、動作制御システムは、図21で説明したように、時相制約処理として支配領域を修正するオブジェクト・ネットワークの準備ができているかどうかをチェックする動作制御を行うことになる。

【0117】準備が完了していることがわかると、支配領域を修正するオブジェクト・ネットワークが起動されるが、支配領域の指定などの制約条件が必要となるので、動作制御システムに必要で制約を要求することになる。動作制御システムは、形態的制約手段として、この制約をディスプレイ上のマネージャ・ウインドウを介してユーザと対話することにより要求し、実行するに必要な制約を供給する動作制御を行うことになる。

【0118】ユーザとサービス・モジュール間のインターフェースとしての役割において、動作制御システムが扱うべきデータを以下に挙げる。ユーザが要求したサービス名（オブジェクトの具体名、オブジェクトネットワーク名、役割名のいずれかでありうる）

ユーザが要求したサービスの実行に必要なデータ（制約）

作業を行っているオブジェクト・ネットワーク名

現在のノード名

当事者（ユーザ、サービス・モジュール）

制約を作成するためのサービス・モジュール名

サーバが実行可能状態であるかを判定するためのフラグ
ユーザが要求したサービスの実行に必要なで一たは十分であるかをチェックするためのフラグ

これらのデータは、動作制御システムが汎用的であるために必要最低限のデータである。それぞれのデータが動作制御システムでどのように使われているかを以下に示す。

【0119】まず、ユーザが要求したサービス名は、ユーザの要求すなわちユーザが行おうとしているサービスを動作制御システムが知るためのデータである。ユーザが要求したサービス名を認識することによって、そのサービスを実行するために必要なデータを、データ管理システムに要求する。このデータはユーザが要求したサービスを実行するのに必要なデータとして十分であるかをチェックするフラグによって、サービス実行可能かどうかのチェックが行われることになる。

【0120】前述したように、チェックの結果、データが不十分であることがわかった場合には、形態的制約処理として、ユーザもしくはサービス・モジュールにより

データを供給または作成してもらう。現在作業を行っているオブジェクト・ネットワーク名と現在のノード名は、サーバが実行できる状態であるかどうかを判定するためのフラグと共に、サーバ上に他の処理を行っていたり、処理が完了していないかどうかをチェックする時相的制約処理を行うために使われる。

【0121】データ当事者は、データを作成する作業者のことで、データ不足の際に当事者にデータを要求して、必要なデータをもらうことになる。制約を埋めるためのサービス・モジュール名は、当事者がサービス・モジュールでデータ不足の場合、データ作成作業を行うことの可能なサービス・モジュールに要求するためのデータである。

【0122】以上のように、動作制御システムは、ユーザとサービス・モジュールとの間で対話をを行い、双方の作業を効果的に行うようにするインターフェースの役割を果たすシステムであることがわかる。また、拡張機能言語Extensible WELL の特性に対応し、主な制御はシステム制約とサービス制約の二つの制約処理によって実現できることなどがわかる。

【0123】以下に於て、役割機能と対話機能について説明する。以下に説明する役割機能と対話機能は、ここまで記述で既に直接的或いは間接的に説明した機能でもあり、必要な部分については上記記述を参照されたい。図23は、役割の定義を示す図である。図23に示されるように、役割は、オブジェクト・ネットワークの構造体として定義され、実行処理として機能する単位である。役割には名前（役割名）が与えられ、その名前によってシステム内外での参照が行われる。

【0124】ある役割の内部に於て、複数のオブジェクト・ネットワーク間の関係は、各オブジェクトネットワークのオブジェクトに対して定義されている制約によって、オブジェクトの属性値間の関係式として規定される。これによって、役割内部の動作が制約処理として実現される。なお役割は、一つのオブジェクト・ネットワークから構成されてもよい。

【0125】本発明のシステムに於て、複数の役割が実行処理を行い、総合的にユーザの意図を満足させるためには、役割間の協調動作が必要となる。そのためには役割間での対話機能を充実させると共に、自由な通信形態を提供する必要がある。またユーザの意図を満足させるためには、ユーザ（支援役割の一つと考えることが出来る）とサービスを行うシステムとの間で、効率的な対話機能を提供する必要がある。前述のように、ユーザとシステムとの間のインターフェース機能を担うのは、コモン・プラットフォームである。このコモン・プラットフォームは、相互に連携している役割間での可視的なインターフェースを提供して、集約された実行状態データを管理すると共に、状況に応じて各サービスモジュールを実行する役割と通信を行う。具体的には、オブジェク

ト・ネットワークがコモン・プラットフォーム上に表示され、それに対して操作を行うと、データや制約を表示するためのデータ・ウインドウが、オブジェクト・ネットワークと同時にコモン・プラットフォーム上に表示される。ユーザ等がデータや制約についての値や性質を与えると、そのオブジェクト・ネットワークが実行処理を行うことによって与えられた役割を果たす。

【0126】本発明のシステムに於ては、ユーザとシステムとの間及び複数の役割同士の間で、効率的な対話機能を提供する必要がある。対話機能としては、以下の様な構成の機能を実現する必要がある。まずevent駆動として、クライアント（ユーザ或いはエージェント・ロール・サーバ）が、コモン・プラットフォーム上の名詞オブジェクトを実現するようにrequestする。コモン・プラットフォームを管理しているシステムがサーバの役割を果たし、その実行結果をrespondする。即ち、ある役割機能が相手の役割機能にサービスを要求するrequestコマンドを発行すると、これを受けて相手の役割機能がサービスを実行処理して、その結果をrespondする。

【0127】またdata駆動として、名詞オブジェクトとして現在扱っているオブジェクトのテンプレート内で属性に対応した値が未定義であるとき、属性値の設定を要求する。なおここで動詞オブジェクトがある総称性を有し、そのパラメータを決定する過程である場合、その動詞オブジェクトは名詞的に取り扱う。属性値の設定を要求する際には、まず属性値が未定義である状況に応答して、データ・ウインドウが表示される。そしてこのデータ・ウインドウ上で、必要とする属性値の定義を、クライアント（ユーザ或いはエージェント・ロール・サーバ）に対して要求する。

【0128】図24は、このようなevent駆動とdata駆動とに基づく対話機能の一例を説明するために、ウェル・システム20（或いは21、22）内部での処理の動きを示す図である。また図25は、図24に対応して、event駆動とdata駆動とに基づく対話機能の一例を示す処理フローである。図24及び図25を参照して、ユーザ（或いはエージェント・ロール・サーバ）がオペレーション・ウインドウ100に表示されたオブジェクト・ネットワークのオブジェクトを指示する（ステップS101）。これはユーザが意図に基づいてrequestを発行したことであり、event駆動に相当する。

【0129】ユーザのオブジェクト指示に対応して、テンプレートが設定される（S102）。いま設定されたテンプレートに於て、対象オブジェクト（目的格等）が未定義であったとすると、テンプレートに於て対象オブジェクトが未定義であることをウェル・システムのカーネル103が判断して、対象オブジェクトの指示を要求する（S103）。これはdata駆動に相当する。

【0130】データ・ウインドウ101に於て、ユーザは対象オブジェクトを指示して、この対象オブジェクト

をテンプレートに代入する（S104）。カーネル103は、テンプレート内部に於て未定義である属性値があるか否かを判断する（S105）。未定義の属性値がある場合には、未定義用データ・ウインドウを表示する（S106）。これはdata駆動に相当する。

【0131】ユーザは、データ・ウインドウ101の未定義用データ・ウインドウに於て、未定義であったデータを定義して（S107）、テンプレートにデータ値を代入する（S108）。ウェル・システムは、実行処理を行ひ、結果をデータ・ウインドウ101に表示する（S109）。これはユーザからのrequestに対して、ウェル・システムがrespondしたことを意味する。

【0132】このようにevent駆動とdata駆動とに基づいた対話機能を提供することによって、ユーザ及びシステム間（ユーザ及びエージェント・ロール・サーバ間）に於てユーザフレンドリで効率の良いインターフェースを実現することが可能となると共に、複数の役割間（エージェント・ロール・サーバとスペシフィック・ロール・サーバとの間等）に於て役割機能間の協調動作を支援する通信機能を実現することが出来る。なお対話機能をWE LLシステムのカーネル部に於て実現することによって、種々のシステム、特にパーソナルコンピュータシステムを考慮したソフトウェアーキテクチャに対応することが出来る。

【0133】また複数の役割間で協調動作を行う際には、ある役割機能を実行している主体である主役割と、その役割実行を支援するためにサービス機能を提供する支援役割とが存在するが、主役割と支援役割との間で、以下のような共通のデータに基づいた対話機能が提供されていることが望ましい。ある役割機能を実行している主役割は、その主役割に關係するある環境下で動作を行っており、この環境に関する環境データを常に監視する必要がある。しかしこのような監視機能を、主役割を支援する支援役割に任せることが出来る。支援役割は環境データを主役割と共有し、環境データに変化のあった時には、主役割に割り込みとして知らせることが出来る。この割り込みに応答して、主役割は環境変化に整合するような動作をすることが可能となる。

【0134】図26は、環境データに基づく主役割機能と支援役割機能との間の対話機能を説明するための図である。図26に於ては、例として2台の自動車を半自動操縦する場合を考える。それぞれの自動車に本システムを組み込んで、互いに衝突するようなコースを走らせるとする。一方の自動車に組み込まれた主役割110は、半自動操縦を行うための操作方法オブジェクトを有しており、この操作方法オブジェクトは、主役割110のコモン・プラットフォーム上でオペレーション・ウインドウ100に表示される。また主役割110のコモン・プラットフォーム上のデータ・ウインドウ101には、環境データが表示される。この環境データは、主役割110

0から支援役割111に転送され共有される。

【0135】環境データが変化して主役割110から支援役割111に転送されると、これをevent駆動として、支援役割111が環境データの特徴的性質を検出する。この特徴的性質の検出は、支援役割111が保持する特徴的性質検出ネットワークによって行われる。図26の例に於ては、例えば、このままで衝突が避けられないほど2台の自動車が接近したという特徴的性質を検出する。支援役割111は、特徴的性質を検出すると、主役割110に対して割り込みを通知する(respondする)。主役割110は、割り込みを受けると、操作方法オブジェクトに対応した動きテンプレートを設定する。この動きテンプレートに未定義部分が存在して、例えば自動車をどの方向にどれだけ移動させるかといったデータが未定義の場合、data駆動によって未定義データの設定を要求する。支援役割111は、未定義データの設定要求に応答し、環境データから必要な特徴的性質を検出して要求されたデータを供給する。このデータが動きテンプレートに代入されると、主役割110は、操作方法オブジェクトを操作ガイドとしてユーザに操作させるために、ユーザとの対話機能を開始する。

【0136】このように主役割の環境に関する環境データを主役割と支援役割とで共有して、主役割と支援役割との間にevent駆動と割り込みとに基づいた対話機能を提供することによって、支援役割が主役割の環境データに対する監視機能を担うこと等が可能になり、主役割が主体として実行する役割機能の動作を支援することが出来る。

【0137】更に複数の役割機能間での協調動作を円滑に行うためには、ある役割を実行する主役割機能から、それに関連する役割を実行する従属的な関係にある従属的役割機能に対して、一对多の放送を行えるようにする必要がある。図27は、主役割機能から従属的役割機能に対する一对多の放送を説明するための図である。

【0138】複数の役割機能の間で協調的な並列実行処理を行っている際に、ある主役割機能の実行中に、それに関連する役割機能に関係のある制御特徴部分が出現したとする。この制御特徴部分を、これら関連する役割機能のコモン・プラットフォーム上の制御特徴部分の有する制御属性値に反映させることによって、関連する役割機能の動作を主役割で発生した制御特徴部分によって制御することが出来る。このようにして、関連する役割機能を、主役割に対する従属的役割機能として位置づけることが出来る。これは主役割の動作について関連役割が興味を持ち、関連役割に於て、主役割に従属して動作するオブジェクトのテンプレートが未定義のデータを含む場合である。

【0139】図27に於て、主役割120と複数の従属的役割123が、システム全体として協調的に動作している。主役割120は、複数の従属的役割123に一対

多くの放送を行うことによって、自らのコモン・プラットフォーム上のテンプレートに指定されている特徴記述項目と同じ項目名についてのデータとして、従属的役割123を制御する。そのために主役割120からのevent駆動に基づいて、支援役割121が特徴制約データを附加した信号を放送する。従属的役割123を支援する支援役割122は、放送された信号を受信して、放送元の役割機能名と特徴制約条件を抽出確認する。従属的役割123は、未定義部分を含むテンプレートを有しており、data駆動に基づいた割り込みとして、支援役割122から特徴制約データを受け取る。受け取った特徴制約データによってテンプレートが定義され、複数の従属的役割123は、主役割120の従属的な役割機能を協調的に実行処理する。

【0140】図28は、役割機能間の通信を説明するための図である。図28に示されるように、役割機能Aと役割機能B及び明示的には図示されない複数の役割機能が、通信環境を介して互いに通信することができる。各役割機能A及びBと通信環境との間には、通信を支援するための通信支援機能が提供される。この通信支援機能は、例えば、図27に於ける支援役割121及び122である。役割機能、通信支援機能、及び通信環境の間は、event駆動とdata駆動とに基づいた対話機能によって通信が行われる。通信支援機能は、通信環境の選択や伝送内容の設定・整合等の動作を行って、役割機能A及びBの通信動作を支援する。

【0141】例えば、役割機能Aに於て、相手役割機能名として役割機能Bを指定して、データ項目名と制約項目名とが指示する内容値が役割機能Bに伝えられる。これによって役割機能Bの実行処理の内容が制御される。このとき、役割機能Bの実行処理に関する制約としては、実行前制約、実行中制約、及び実行後制約の各制約条件が設定されて、協調動作に於ける整合性をチェックする。複数の役割機能間に於て、通信対象は特定の相手に限られることなく、自由に相手を選択して通信することが可能である。

【0142】

【発明の効果】以上説明したように、本発明によれば、モデルとして、データ・モデル、オブジェクト・モデル、役割モデル、プロセス・モデルを設定し、これらのモデル間の結合が制約を中心とした規則によって行われることにより、モデル駆動が実行処理過程を規定し、ユーザフレンドリなシステム使用およびエキスパートによるシステム設計を容易に実現できるようになる。

【0143】また、データと制約とをシステム内で同一の処理方法によって処理し、ユーザ、エージェント・ロール・サーバ、スペシフィック・ロール・サーバ間で、データと制約との授受によりプロセスを進行させることによって、データや制約の統一的な管理が可能になり、ユーザとサービス・モジュールとの間で対話をを行い、双

方の作業を効果的に行うように導くインターフェースを提供することができるようになる。

【0144】また、システムの制約をオブジェクト・ネットワーク内でのオブジェクト間関係によって規定し、それによって制約処理の方法を制御することにより、単純な方法によって複雑なサービスを実現することが可能になる。エキスパートがサービス実行のためのオブジェクト・ネットワークを設計するときに、システムの持つモデルに基づいて制約を利用することができるので、システムの開発を効率よく進めることができる。

【0145】また、エージェント・ロール・サーバを階層化し、上位と下位のエージェント・ロール・サーバの関係を、エージェント・ロール・サーバとスペシフィック・ロール・サーバとの関係と同様にすることにより、複雑なサービスを単純な構造によって実現し、システム全体としての協調処理の制約的実現が可能になる。

【図面の簡単な説明】

【図1】本発明の構成例を示す図である。

【図2】実世界と感性的対象世界との連環性を示す図である。

【図3】画像についての形式モデルである幾何モデルの例を示す図である。

【図4】特徴モデルの構造を説明するための図である。

【図5】実行処理の進行の例を示す図である。

【図6】色度、輝度勾配、輝度値を入力するためのデータ・ウインドウの例を示す図である。

【図7】テンプレートの例を示す図である。

【図8】システム制約とWE LLとの関係を示す図である。

【図9】実行時システム制約(AND制約)による動作制御システムに関する処理の流れを示す図である。

【図10】実行時システム制約(OR制約)による動作制御システムに関する処理の流れを示す図である。

【図11】階層制約を説明するための階層の例を示す図である。

【図12】階層制約処理の流れを示す図である。

【図13】オブジェクトの接続順序の例を示す図である。

【図14】動詞オブジェクトのテンプレートとデータスキーマの例を示す図である。

【図15】制約処理と内包論理を説明する図である。

【図16】階層化されたエージェント・ロール・サーバの例を示す図である。

【図17】領域の移動に伴って画像の不連続な部分が生

じる例を示す図である。

【図18】画像の階層制約処理の例を示す図である。

【図19】動画像描画システムにおけるサービス制約の一例である連続性制約に関する対話的処理の流れを示す図である。

【図20】動作制御システムの制御の流れを示す図である。

【図21】動作制御システムの制御の流れを示す図である。

10 【図22】動作制御システムによる制御の例を示す図である。

【図23】役割の定義を示す図である。

【図24】even駆動とdata駆動に基づく対話機能の一例を説明するためにウェル・システム内部での処理の動きを示す図である。

【図25】図24に対応して、event駆動とdata駆動に基づく対話機能の一例を示す処理の流れを示す図である。

20 【図26】環境データに基づく主役割機能と支援役割機能との間の対話機能を説明するための図である。

【図27】主役割機能から従属的役割機能に対する一対多の放送を説明するための図である。

【図28】役割機能間の通信を説明するための図である。

【図29】画像描画処理用のオブジェクト・ネットワークの例を示す図である。

【符号の説明】

10 ユーザ(クライアント)

20 ウエル・システム(エージェント・ロール・サーバ)

30 21, 22 ウエル・システム(スペシフィック・ロール・サーバ)

30, 31, 32 コモン・プラットフォーム

40, 41, 42 動作制御システム

50, 51, 52 データ管理システム

100 オペレーション・ウインドウ

101 データ・ウインドウ

103 カーネル

110 主役割

40 111 支援役割

120 主役割

121 支援役割

122 支援役割

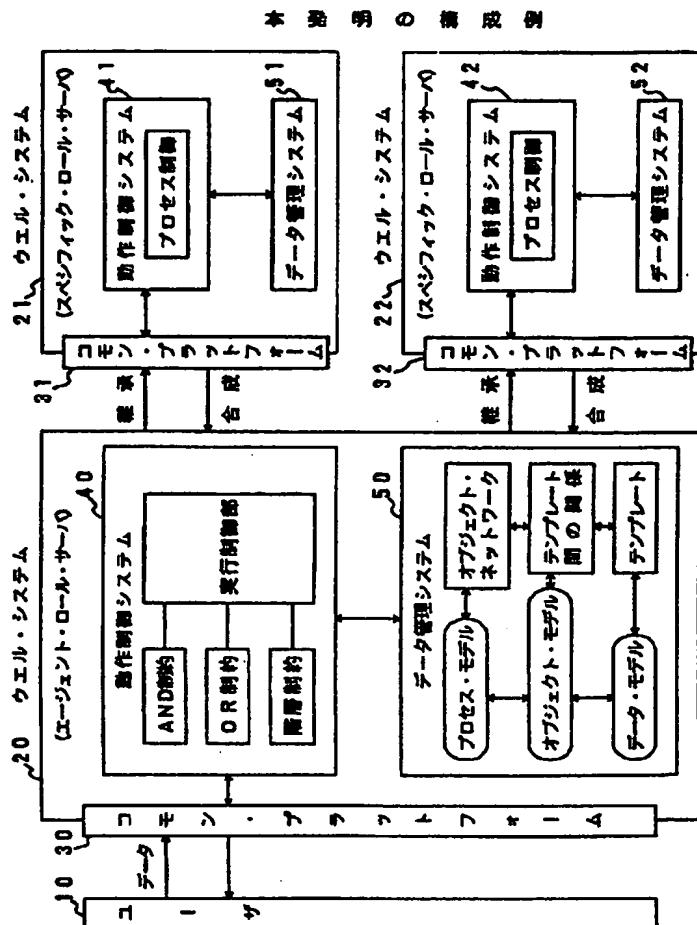
123 従属的役割

【図7】

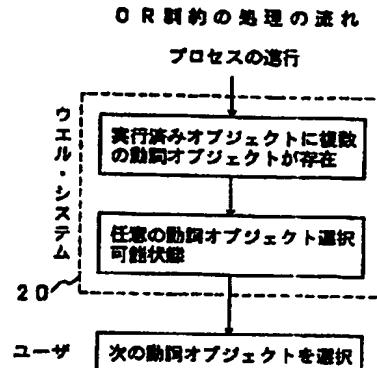
テンプレートの例

ID	名前	X	Y	L	Lx	Ly	Cx	Cy	ポインタデータ
----	----	---	---	---	----	----	----	----	---------

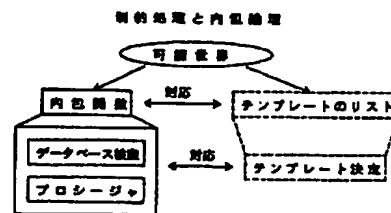
【図1】



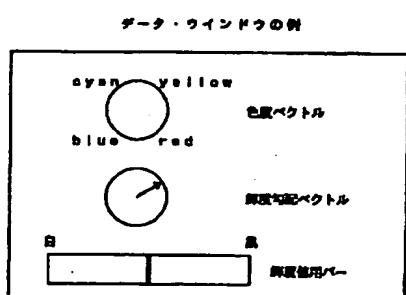
【図10】



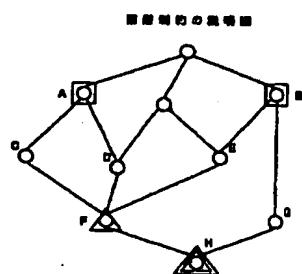
【図15】



【図6】



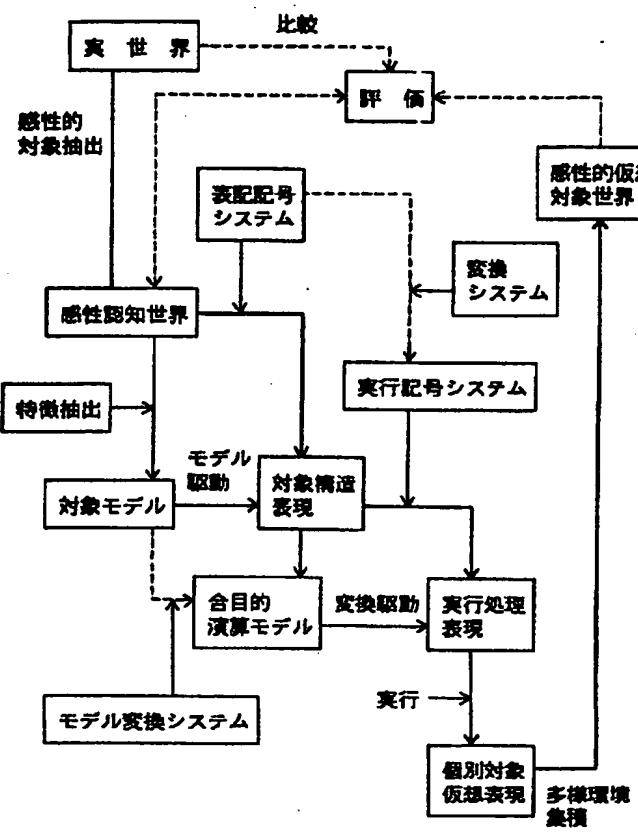
【図11】



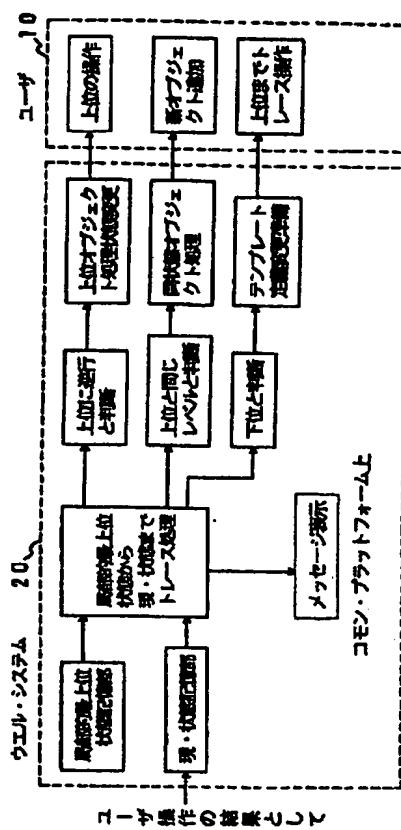
【図2】

【図12】

実世界と感性的対象世界との連理性



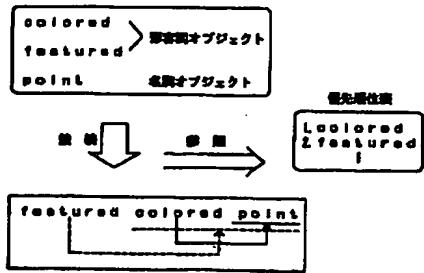
階層制約処理の流れ



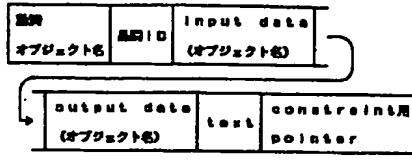
【図13】

【図14】

オブジェクトの並列順序



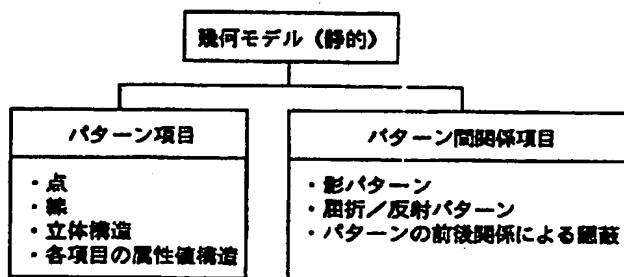
面倒オブジェクトのテンプレートとデータスキーマ



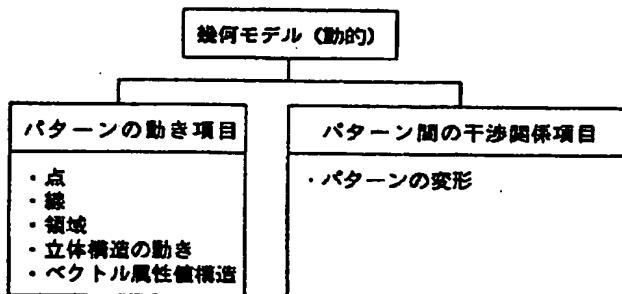
【図3】

形式モデル（幾何モデル）の例

(A)



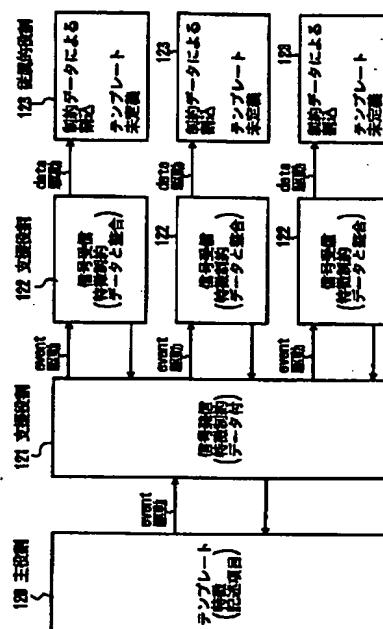
(B)



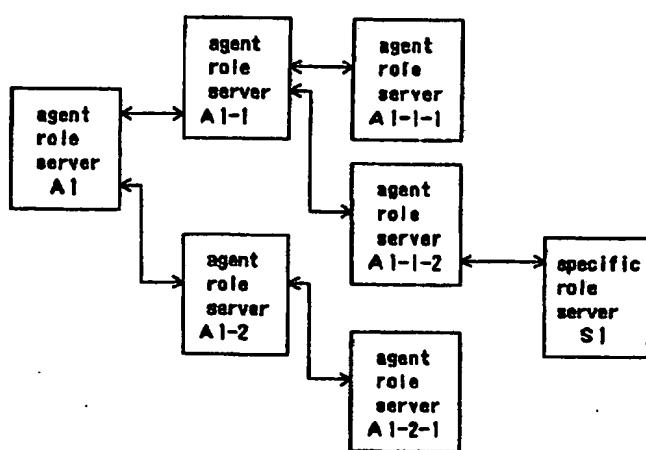
【図16】

【図27】

主投票実施局から従属的投票実施局に対する
一太多の放送を説明するための図

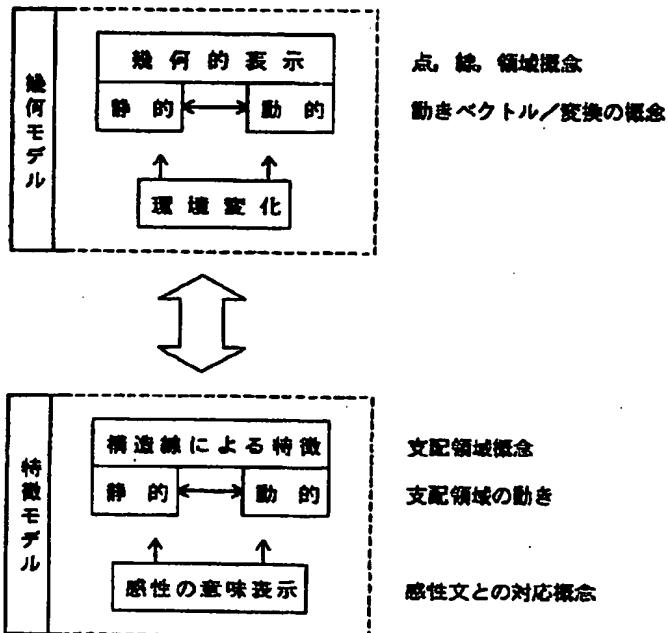


階層化されたエージェント・ロール・サーバ



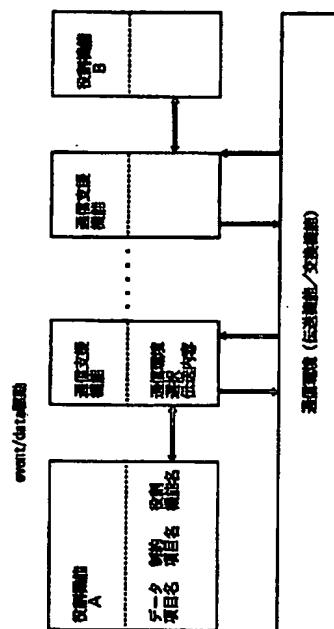
【図4】

特徴モデルの構造



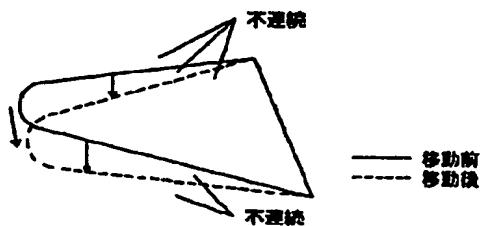
【図28】

役割実現時間の適度を説明するための図

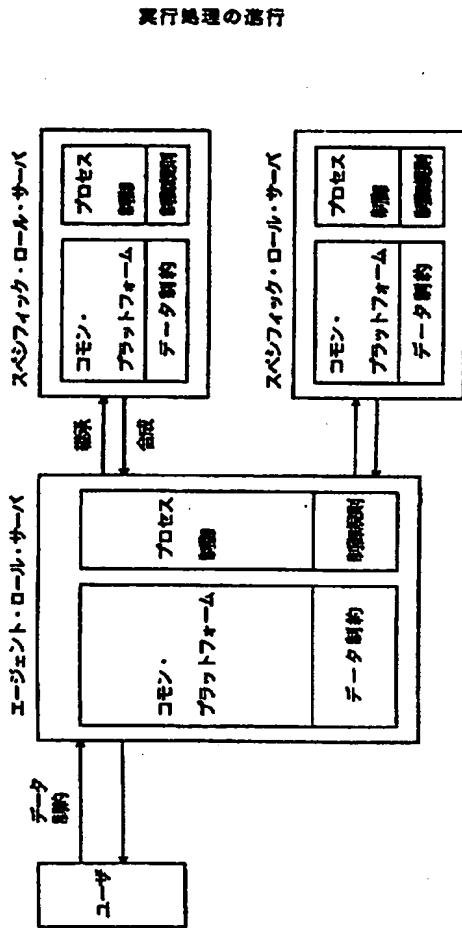


【図17】

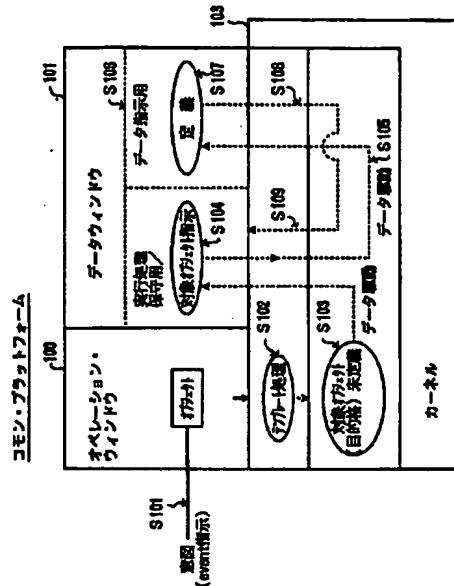
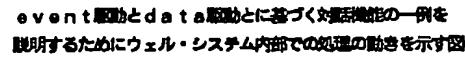
領域の移動に伴う画像の不連続



[図5]



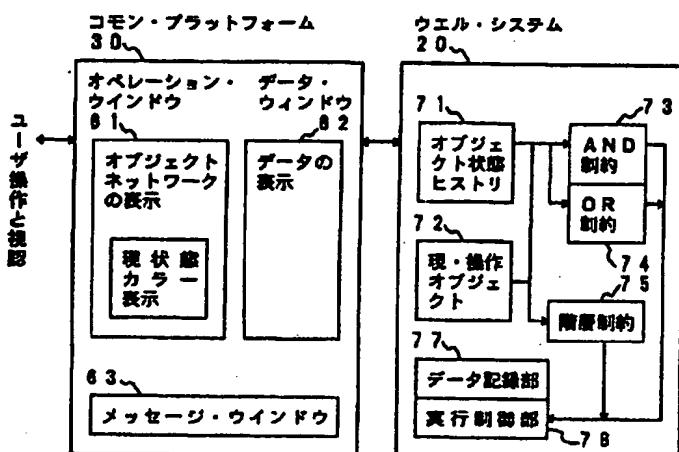
[図24]



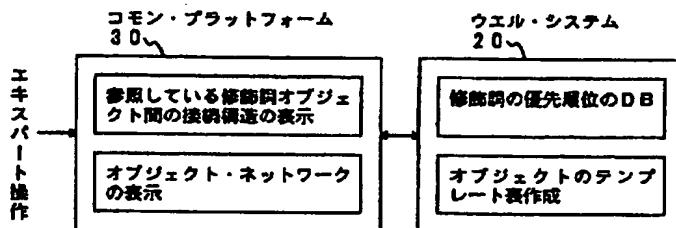
【図8】

システム制約とWELLとの関係

(A) 実行時システム制約における実行制御

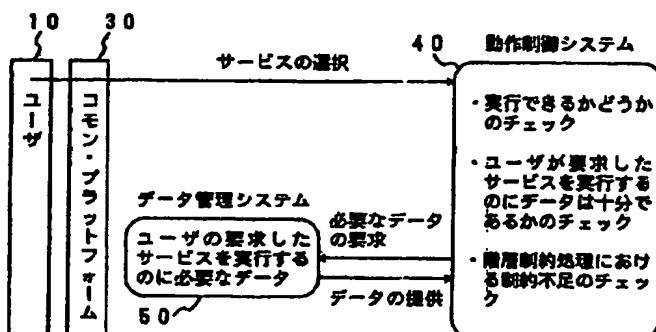


(B) オブジェクト・ネットワーク設計時の制約規則による対話



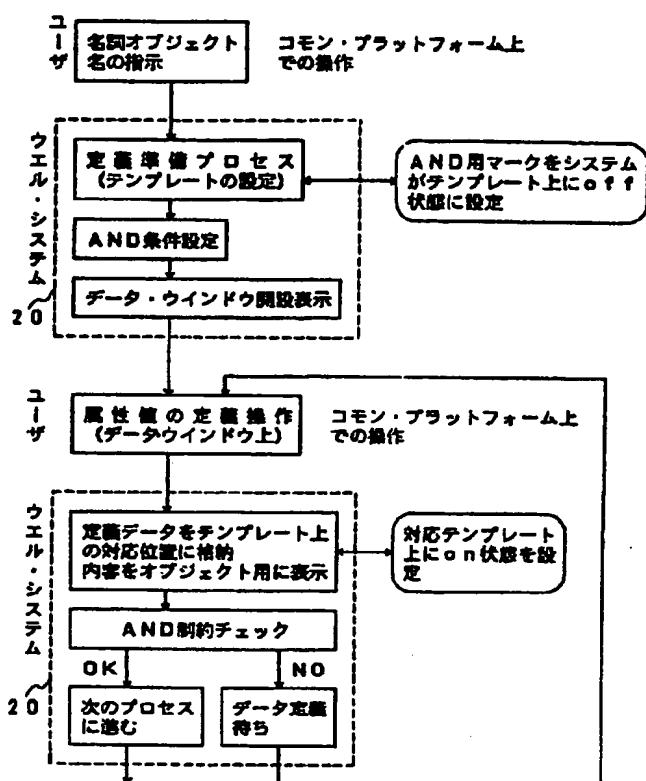
【図20】

動作制御システムの制御の流れ (1)

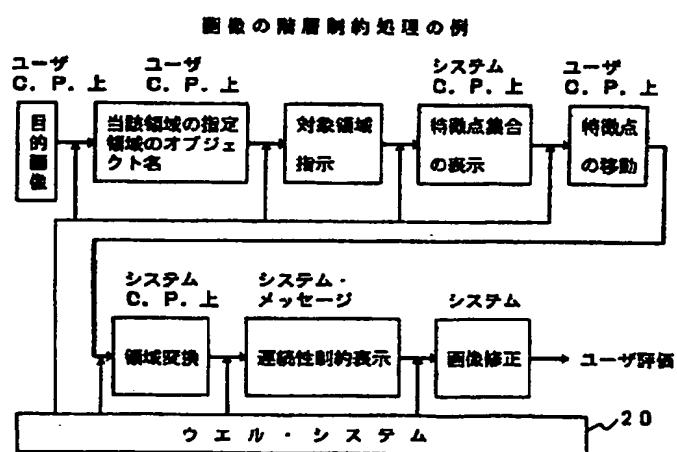


[図9]

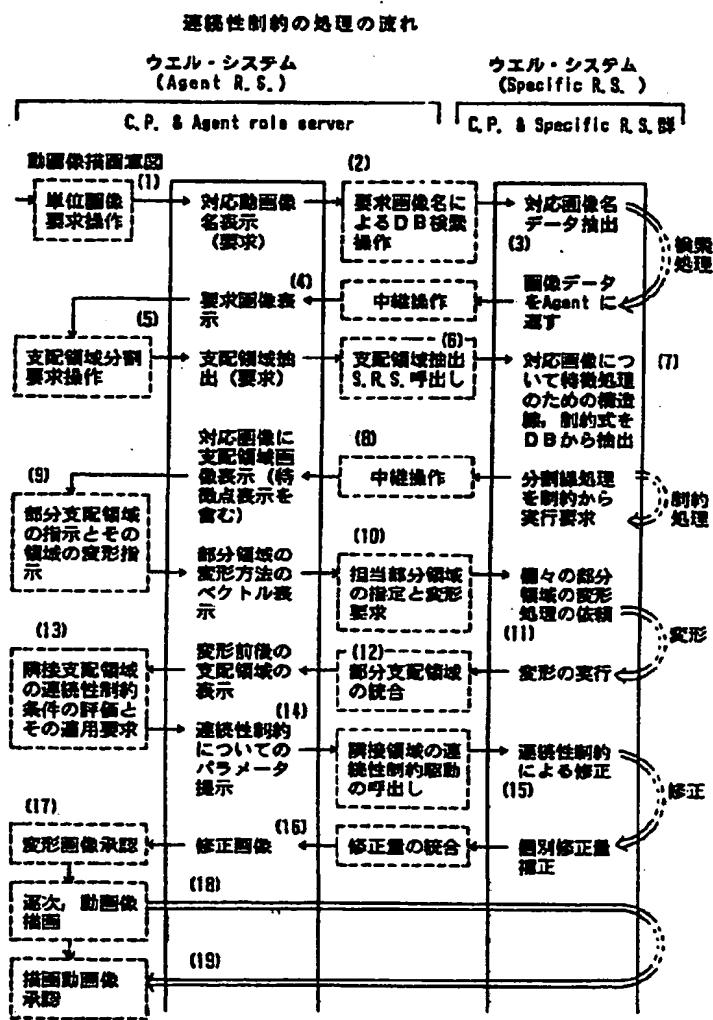
AND制約の処理の流れ



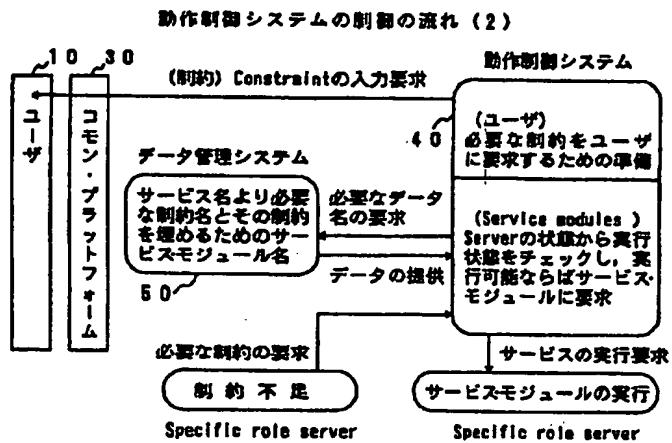
[図18]



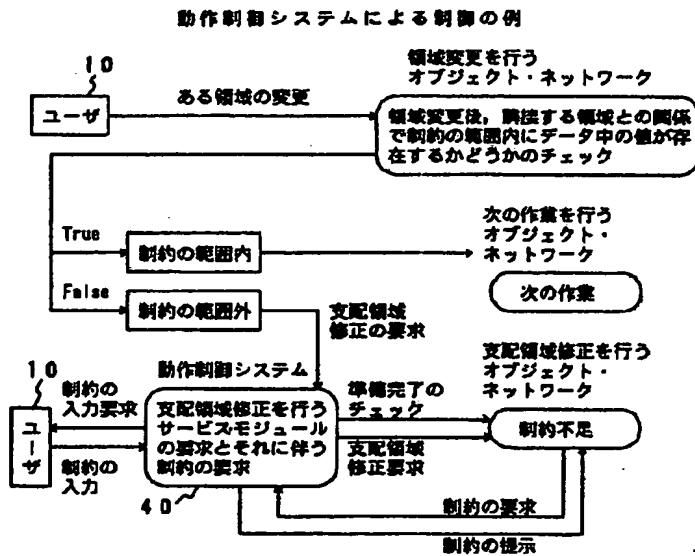
【図19】



【図21】

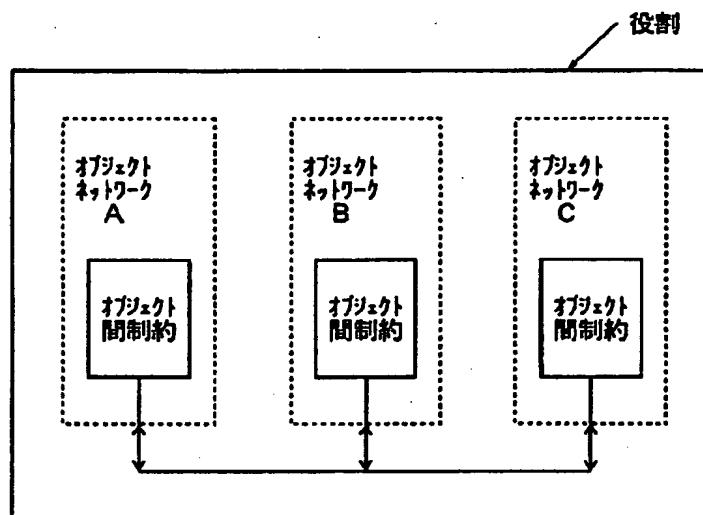


【図22】



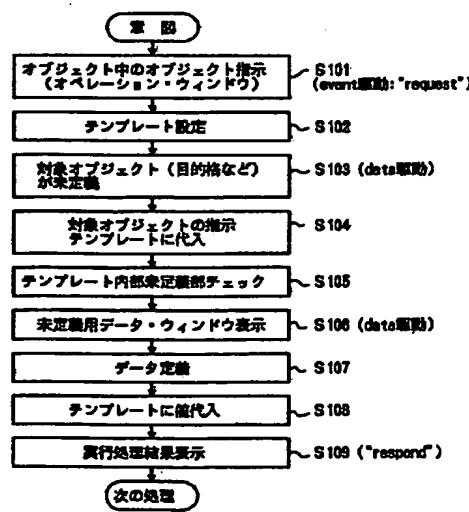
【図23】

役割の定義を示す図



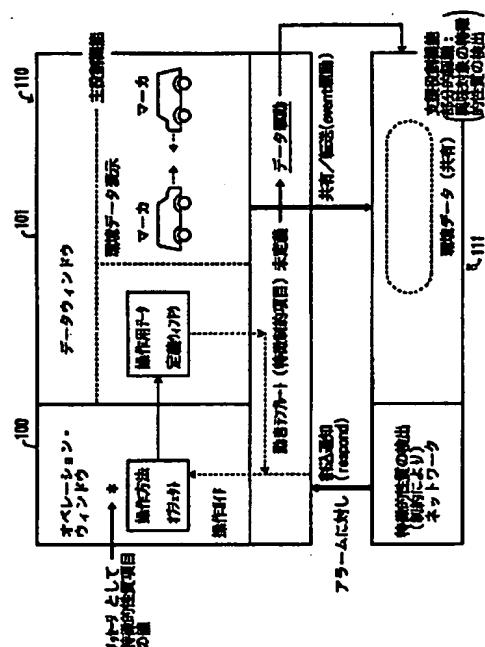
【図25】

図24に対応して、event駆動とdata駆動とに
基づく対話機能の一例を示す処理の流れを示す図



【図26】

環境データに基づく主役割機能と支援役機能との
間の会話機能を説明するための図



【図29】

画像描画処理用のオブジェクト・ネットワーク

